VYSOKÁ ŠKOLA BÁŇSKÁ – TECHNICKÁ UNIVERZITA OSTRAVA
EKONOMICKÁ FAKULTA

KATEDRA SYSTÉMOVÉHO INŽENÝRSTVÍ

Návrh a implementace open source BI řešení pro IT firmu
Design and Implementation of Open Source BI Solution for the IT Company

Student: Bc. Tomáš Tomis
Vedoucí diplomové práce: doc. Ing. Milena Tvrdíková, CSc.

Ostrava 2016

VŠB - Technická univerzita Ostrava
Ekonomická fakulta
Katedra systémového inženýrství

Zadání diplomové práce

Student: **Bc. Tomáš Tomis**

Studijní program: N6209 Systémové inženýrství a informatika

Studijní obor: 6209T025 Systémové inženýrství a informatika

Téma: Návrh a implementace open source BI řešení pro IT firmu

Design and Implementation of an Open Source BI Solution for the IT Company.

Jazyk vypracování: čeština

Zásady pro vypracování:

1. Úvod
2. Teoretická východiska Business Intelligence zaměřená na získávání dat
3. Výběr a analýza Open Source Business Intelligence nástrojů
4. Testování zúženého výběru dle firemních požadavků a jeho vyhodnocení
5. Implementace nejvhodnějšího řešení
6. Závěr

Seznam použité literatury
Seznam zkratek
Prohlášení o využití výsledků diplomové práce
Seznam příloh
Přílohy

Seznam doporučené odborné literatury:

BIERE, Mike. *The New Era of Enterprise Business Intelligence: Using Analytics to Achieve a Global Competitive Advantage.* Boston: Pearson Education, 2011. ISBN 978-0137075423.
HOWSON, Cindi. *Successful Business Intelligence: Secrets of Making BI a Killer App. New* York: McGraw-Hill, 2008. ISBN 978-0071498517.
Pour, Jan, Miloš Maryška a Ota Novotný. *Business Intelligence v pdnikové praxi.* Praha: Professional Publishing, 2012. ISBN 978-80-7431-247-065-2.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **doc. Ing. Milena Tvrdíková, CSc.**

Datum zadání: 20.11.2015

Datum odevzdání: 22.04.2016

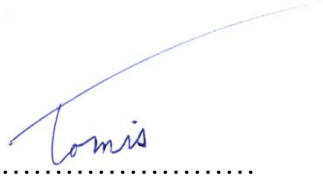
doc. Ing. Jana Hančlová, CSc.
vedoucí katedry

prof. Dr. Ing. Dana Dluhošová
děkanka fakulty

Na tomto místě bych chtěl poděkovat doc. Ing. Mileně Tvrdíkové, CSc., za výbornou spolupráci, užitečné rady a profesionální přístup při vedení mé diplomové práce.

„Prohlašuji, že jsem celou práci, včetně všech příloh, vypracoval samostatně“.

V Ostravě dne 22. dubna 2016



.....

Tomáš Tomis

Adresa trvalého pobytu studenta:

Na Sedlácích 1011

739 34 Šenov

Obsah

1. Úvod

Kvalitní data jsou již několik desítek let jedním ze stěžejních faktorů, které mohou výrazně změnit konkurenceschopnost a firemní postavení na trhu, obzvláště pokud jsou strukturovaná. Firmy se naučily vnímat jejich důležitost a hodnotu, snaží se rozšiřovat obzory ve svém odvětví, hledat nové souvislosti a využívat je ve svůj prospěch. Je samozřejmostí, že skoro každá firma, obzvláště pak v IT sféře, má dnes vlastní databázi obsahující důležitá firemní data. Především se jedná o zákaznická data, ale také jiné typy dat podle povahy firmy a odvětví, ve kterém se pohybuje.

S raketovým rozmachem informačních technologií, který odstartoval koncem devadesátých let a úspěšně se přenesl i do nového milénia, se rapidně zrychlil nejen životní styl, ale také celkový přístup k podnikání a jeho podpoře v IT.

Díky celoplanetární globalizaci, médiím, sociálním sítím a moderním komunikačním technologiím vznikají denně obrovské objemy dat jak ve firemním prostředí samotném, tak i mimo něj. Většina firem se snaží tyto data zpracovávat a využít je ve svůj prospěch. Nicméně vzhledem k jejich množství a rychlosti jejich vytváření je to velmi složitý úkol. Vznikají technologie jako framework Hadoop, které jsou určeny ke zpracování takto velkého objemu nesourodých a často nestrukturovaných dat.

Pro vedení firmy se intenzivní využívání moderních IT technologií stalo jednou z hlavních součástí procesu řízení firmy. Manažeři musí rychle reagovat na změny v podniku a jeho okolí. Díky novodobým přístrojům, jako jsou chytré telefony a tablety, v kombinaci s bezproblémovým připojením k internetu na většině míst moderního světa, mohou vedoucí pracovníci neustále sledovat výsledky svého týmu, oddělení nebo celého podniku.

Reporting, vizualizace dat a business intelligence vůbec se staly velmi žádanými a profitujícími oblastmi. Větší společnosti v oblasti informačních technologií dokonce často mívají celá oddělení, která se věnují nástrojům pro zvyšování kvality dat, datové analýze a vizualizaci. Pro vedoucí pracovníky bez expertních znalostí z každého odvětví je mnohem jednodušší najít výsledky nebo sledovat růst firemního výkonu v logicky sestavené grafické formě, než jen sledovat nepřehledné tabulky se spoustou čísel. Manažeři si přejí software, který umožní velice jednoduchou a intuitivní ad hoc analýzu, tvorbu vlastních dashboardů, klíčových indikátorů výkonnosti (KPI) a grafů. To vše jen pomocí několika jednoduchých operací.

S nadsázkou lze říct, že se sledování výkonu a dělání důležitých rozhodnutí přesunulo z častých schůzek do online meetingů. To vše díky využívání chytrých přístrojů a sledování výkonu v reálném čase. Potřebná data o výkonnosti jsou do firemní databáze nahrávána neustále. Na firemním serveru neustále běží reportingová aplikace, ve které jsou vytvářeny dynamické reporty, dashboardy a grafy. Takto vytvořené prvky mají vlastnost velmi častého obnovování, takže prezentují vždy nejnovější data a ideálně v reálném čase. Tato vlastnost následně umožňuje velmi rychlou reakci vedení.

Opakovaná tvorba statických reportů a grafů z nových dat zbytečně bere čas vývojářům a ještě k tomu jsou reporty dodány se zpožděním. Proto je varianta dynamických dashboardů, grafů a reportů, které běží přímo nad samotnou databází a bez dalšího zásahu vždy vyobrazí nová data, výrazně efektivnějším řešením.

Cílem diplomové práce je navrhnout a realizovat pro firmu business intelligence řešení se zaměřením na dynamický reporting. Jedním z důležitých faktorů je, že toto řešení má být pokud možno s minimálními náklady, proto bude v řešení využito open source nástrojů.

První část je věnována teoretickým východiskům pro oblast business intelligence, nástrojům pro získávání a zvyšování kvality dat, datovým uložištím, reportingem a vizualizací dat.

Poté bude následovat praktická část, ve které jsou zhodnoceny možnosti na poli open source business intelligence nástrojů, dále je vybráno nejlepší řešení pro náš konkrétní projekt a celé toto řešení je implementováno dle požadavků zákazníka.

V závěru je proveden souhrn řešení a uvedeny poznatky, které byly získány realizací tohoto řešení.

2. Teoretická východiska Business Intelligence zaměřená na získávání dat

V této části se budeme zabývat teoretickými východisky z oblasti business intelligence, definováním tohoto pojmu, jeho složkami a oblastmi, kterých se dotýká. Budeme se také blíže zabývat pojmy jako ETL (extraction, transformation, loading), ELT (extraction, loading, transformation) a jejich vzájemnými rozdíly. Dále nastíníme, co jsou to datová uložiště a jak se s nimi pracuje. Kapitolu zakončíme oblastí reportingu a vizualizace dat.

2.1. Business Intelligence

Co je business intelligence? To je fundamentální otázka celé této problematiky. Business intelligence je pojem, který nemá svou ustálenou definici. Existuje však mnoho různých variant této definice, které by dle známého odborníka v oblasti BI Davida Loshina, bylo možno sumarizovat asi takto.

Je to souhrn nástrojů, procesů a technologií, které slouží k postupné transformaci dat v informace a informací ve znalosti. Znalosti takto získané jsou následně využívány k podnikovému plánování a efektivnímu rozhodování, což vede k úspěšnosti celého podniku. Celá oblast zahrnuje práci s datovými sklady, znalostní management a podnikatelské analytické nástroje. (Loshin, 2012)

BI systém by měl ovšem zahrnovat mnohem více než jen software pro práci s daty, reportingový software, analytické nástroje a software pro vizualizaci dat. Důležité je, že všechny tyto prostředky by měly konečnému uživateli (většinou manažerovi) poskytovat dostatek využitelných informací, díky kterým může rychle a efektivně rozhodovat a reagovat. Stejně jako je důležitá samotná schopnost daného člověka umět správně zareagovat a znalost procesů k tomu potřebných. Z toho vyplývá, že bez správných procesů a lidí, kteří se nebojí rozhodovat včas, jsou BI nástroje takřka nevyužitelné. Takže efektivnost BI musí být vždy sledována v kontextu s reakcí lidí ve vedení. (Biere, 2011)

Business intelligence můžeme také zjednodušeně definovat jako soubor matematických modelů a metodologií analýzy, které využívají veškeré dostupné údaje pro zisk informací a znalostí. Nabyté znalosti jsou, jak už bylo řečeno výše, poté užitečné pro komplexní rozhodovací procesy. Důležitým aspektem celého BI je také propojenost s jinými obory a identifikace primárních složek, které jsou pro business intelligence typické v těchto oborech. (Vercellis, 2009)

V posledních letech byl zaznamenán rozvoj v podnikatelské sféře zejména díky vzniku nízkonákladových technologií pro ukládání dat a širokou dostupnost připojení k Internetu. Díky tomu je nejen pro jednotlivce, ale pro celé organizace podstatně snazší přistupovat velmi rychle ke gigantickému množství dat. Tato data jsou však často velmi různorodá, co se týče jejich obsahu a forem jejich reprezentace. Mohou totiž obsahovat finanční a administrativní transakce, elektronickou korespondenci, data ze sociálních sítí, textové soubory, hypertextové soubory a mnoho dalších dat různého typu a povahy. Jejich dostupnost však firmě otevře dveře k slibným scénářům a možnostem. První myšlenka, která by racionálně uvažujícímu vedení firmy měla přijít na mysl je, zda je možné transformovat všechna tato data do informací a znalostí. Pak by mohla být využita k podpoře a zlepšení řízení firmy. (Vercellis, 2009)

V této situaci nastupuje business intelligence řešení, které se firma bude snažit implementovat pro zajištění zisku těchto leckdy klíčových a potřebných informací, potažmo znalostí.

2.1.1. Data, informace, znalosti

Jde o tři, již několikrát zmíněné, pojmy, kolem kterých je teorie nástrojů business intelligence postavena. Často jsou však bohužel zaměňovány a kombinovány dohromady. Proto bychom měli stanovit pevnou definici těchto tří pojmů, aby byly rozdíly mezi nimi na první pohled evidentní, byly dobře pochopeny a nedocházelo k záměnám.

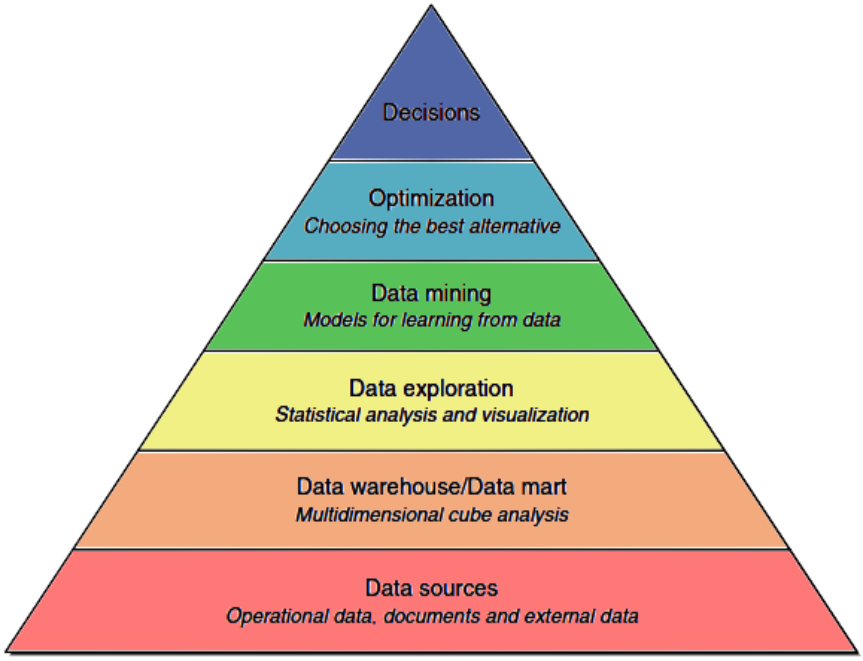
- Data – Jsou kolekcí hodnot získaných v „surové“ nezpracované formě. Využívají se dále k výpočtům a měřením. Jsou obrazem skutečnosti bez jakýchkoliv úprav a můžeme z nich vyvodit jakýkoliv důsledek. (Loshin, 2012)
- Informace – Jsou výsledkem transformace a zpracování dat. Jedná se o uspořádání a vytvoření vztahů mezi daty. Dávají smysl především odborníkům z dané oblasti. (Vercellis, 2009)
- Znalosti – Znalosti vzniknou z informací tím, že je využijeme k tvorbě rozhodnutí a k provedení akcí souvisejících s tímto rozhodnutím. Takže můžeme za znalosti považovat konzistentní informace, které jsou prakticky využity v příslušné oblasti, díky zkušenostem a kompetencím člověka, jež rozhodnutí učiní. (Vercellis, 2009)

Takto by se tedy daly definovat všechny tři výše zmíněné pojmy. Jelikož jde o jedny ze základních pojmů BI, jsou často uváděny v rámci jiných částí práce. Naše úvahy budou vždy vycházet z této definice pojmů.

2.1.2. Architektury Business Intelligence

„*Imagine having the humanoid robot Data from Star Trek: The Next Generation as a consultant: A massive database that can sift through all the data ever known and turn it into information that the captain (okay, CEO) can use to make decisions... all in a human-friendly package. After he's spent a few days plugging in to all your computer systems and interviewing your people, you can ask him any question about your organization — and get a useful answer. The perfect business intelligence solution!*" (Withee, 2010)

Architektura systému pro business intelligence řešení bývá často patřičně rozsáhlá. Nicméně když budeme celý systém dostatečně generalizovat, dal by se rozdělit na tyto hlavní části. Zdroje dat, datové sklady a datová tržiště, která jsou infrastrukturou celého systému, metodologie samotného business intelligence, prozkoumávaní dat a jejich analýza, dolování dat, optimalizace a rozhodování. Všechny výše zmíněné části budou popsány a vysvětleny v následujících bodech a jejich vzájemnou návaznost můžeme vidět v obrázku 2.1.



Obrázek 2.1. Pyramida BI komponent, zdroj: (Vercellis, 2009)

Datové zdroje

Primární a sekundární zdroje pro zisk dat často mívají velice různorodý typ a formu. Je třeba je co nejefektivnějším způsobem unifikovat a spojit dohromady. Obsahují převážně data z podnikových systémů, ale mohou také obsahovat dokumenty a data získaná od externích poskytovatelů nebo například ze sociálních sítí. (Vercellis, 2009)

Datové sklady a datová tržiště

Slouží k ukládání dat, které získáme z ETL procesů. Data, která byla původně z různých zdrojů, se takto ukládají do databáze. Tato data jsou následně využívána k případným analýzám a vizualizacím pomocí reportingových a analyzačních nástrojů. Poté mohou být využita pro podporu BI. (Loshin, 2012)

Metodologie BI

Když jsou data konečně extrahována z primárních systémů, jsou podrobena různým analytickým metodám. Data také bývají „proháněna“ matematickými modely, které by měly napomoci tvorbě rozhodnutí vedení. Často bývá součástí systémů několik druhů aplikací na podporu rozhodování a jejich využívání se liší podle firmy a přístupu. Mezi takovéto přístupy patří například analýza multidimenzionální kostky, induktivní učící modely dolování dat, datová analýza, analýza časových řad, optimalizační modely a mnoho dalších. (Vercellis, 2009)

Průzkum dat

Jedná se především o pasivní metodologie průzkumu dat, kdy si např. někdo z vedení firmy všimne, že poklesly zisky za určité období. Poté se pomocí extrahování dat, jejich následné vizuální reprezentace a také pomocí testování statistických hypotéz snaží svůj závěr potvrdit, či vyvrátit. Pasivní metodologie průzkumu dat jsou často složeny z dotazovacích a reportovacích systémů. (Vercellis, 2012)

Dolování dat

Dolování dat neboli data mining je na rozdíl od průzkumu dat založeno na aktivních průzkumových metodologiích. Jeho smyslem není ověřovat, zda reálná situace skutečně vychází i z dat, které jsou o ní doloženy, nýbrž se snaží získávat z dat nové souvislosti, informace a znalosti. Tato metodologie zahrnuje matematické modely, které slouží pro rozpoznávání vzorů a souvislostí v datech, dále také může obsahovat techniky strojového učení. Nově získané souvislosti mohou poskytnout kýženou konkurenční výhodu nebo rozšířit znalosti a tím i usnadnit budoucí rozhodování. (Vercellis, 2012)

Optimalizace

Optimalizací se rozumí vybírání té nejlepší alternativy ze všech řešení. Množina řešení bývá často velmi rozsáhlá (někdy dokonce takřka nekonečná), takže se rozhodně nejedná o jednoduchou činnost.

Rozhodnutí

Rozhodnutí je pomyslnou první příčkou v procesu business intelligence a jak můžeme vidět na obrázku 2.1. je na samém vrcholu BI pyramidy. Jedná se o reprezentaci přirozeného shrnutí poznatků a dosavadně získaných informací, které jsou následně využity pro ustanovení výsledného rozhodnutí. I když se povede úspěšně zakomponovat metodologie business intelligence, tak to nemusí vždy znamenat, že budou vznikat ta nejlepší rozhodnutí. Finální fáze rozhodování je totiž ve velké míře závislá na lidském faktoru. Jinými slovy, vrcholoví manažeři nebo vedení firmy může rozhodnout i bez jakéhokoliv ohledu na výsledky testů, či analýz. (Vercellis, 2012)

Když se podíváme na celou pyramidu z obrázku 2.1. jako celek, můžeme vidět, že od samého startu až do konce neustále narůstá potřeba podpůrných nástrojů pro každou úroveň. Čím výše se v pyramidě dostaneme, tím více narůstá potřeba aktivních nástrojů BI.

Stejně tak se mění i kompetence a role uživatelů systému. V nižších úrovních se může jednat o práci specialistů na informační systémy nebo databázových administrátorů. Ve středních fázích se kompetence přesouvají směrem k analytikům a expertům na matematické modelování a statistické metody. Na vrcholu toho všeho stojí pracovníci, kteří rozhodují v podobě zainteresovaných manažerů nebo vedení firmy.

Business intelligence systémy a řešení se tedy starají o potřeby různých typů složitých a mnohdy velmi komplexních organizací, které mívají často obchodní, průmyslový nebo IT charakter. BI systémy můžeme najít převážně v odděleních firemní logistiky, produkce, marketingu, prodeje, účetnictví nebo například v oblasti kontroly a monitoringu. (Biere, 2011)

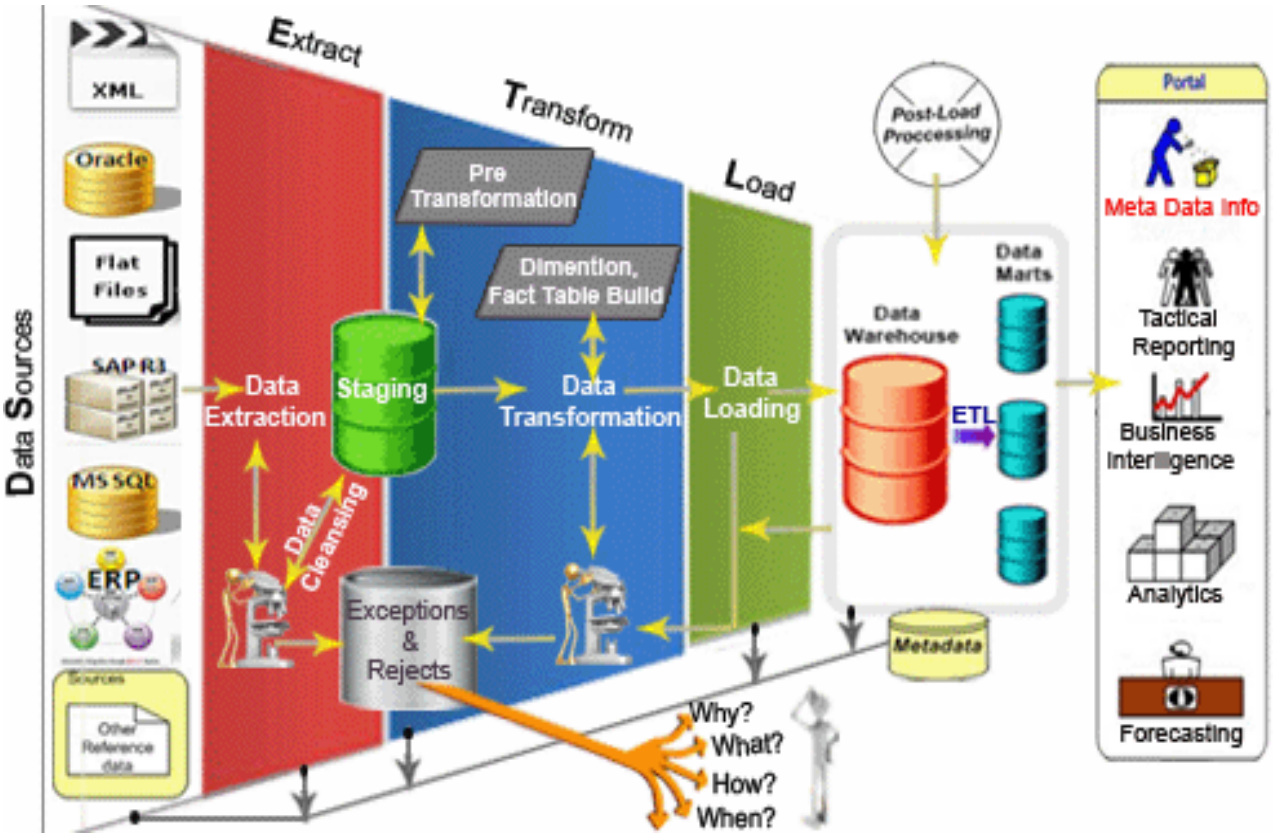
2.2. ETL

Jde o jednu z nejdůležitějších částí BI řešení. ETL je obecně uznávaná zkratka pro činnosti extraction, transformation, loading (extrakce, transformace, nahrávání). Jejich názvy už poměrně jasně indikují, oč se jedná. Extrakce slouží k získání dat, transformace k následné úpravě dat do potřebné podoby a nahrání slouží k uložení dat do datového skladu. Jedná se o propojený řetězec kroků, kde na sebe jednotlivé kroky navazují. Často pracují v takzvaných batch intervalech, což jsou časově oddělené dávky. Většinou se jedná o denní, týdenní nebo také měsíční intervaly.

V praxi jde o zisk dat z primárních systémů (například data získaná z čteček nebo různých čidel používaných ve výrobě). Tato data se následně extrahují a ukládají do tzv. DSA (Data

Staging Area), což je v podstatě uložiště, kde získaná data čekají na transformaci. Následně jsou data z DSA transformovány a nahrány do datového skladu.

Pro tyto úkony existuje nespočet ETL nástrojů od freeware open source nástrojů až po vysoce specializované drahé nástroje jako například Oracle Warehouse Builder, Microsoft SSIS, PowerCenter Informatica, TOS for Data Integration a mnoho dalších.



Obrázek 2.2. Schéma ETL a jeho částí, zdroj: (DiKube, 2011)

Na schématu z webu firmy DiKube můžeme vidět podrobně graficky rozpracované části celého ETL procesu. Je také patrná návaznost reportingu, tvorby předpovědí a analýz na samotný proces ETL. Jednotlivým částem se budeme věnovat v následujících bodech.

2.2.1. Extrakce

Extrakt se do češtiny překládá jako výtažek, což logicky naznačuje, že při extrakci se bude jednat o „vytahování“ dat. Slouží tedy k získávání dat z různých typů primárních systémů.

„The first step of integration is successfully extracting data from the primary source systems.“ (Kimball, 2013)

Během svého vývoje podnik získává a dědí velké množství různých infrastruktur a počítačových systémů, sloužících k chodu podniku. Tyto systémy se mohou týkat různých oblastí jako správa inventáře, kontrola produkce atd. Bohužel tato věcná a časová různorodost způsobuje nejen fyzický, ale také logický nesoulad mezi jednotlivými systémy. Mohou se lišit v hardware, operačním systému nebo i databázovým systémem samotným. (Kimball, 2013)

Proto je nutné data extrahovat, aby bylo možno je uložit do nově navržené logické struktury, která zachová vztahy mezi původními a cílovými daty. Jedná se tedy o periodickou činnost, jež je neustále opakována a plyne z ní zisk nových dat. Nicméně většinou nelze data rovnou za běhu transformovat a nahrávat přímo do finálního datového skladu. Proto vzniká „mezi uložiště“, které je v praxi nazýváno Data Staging Area.

Jak už bylo řečeno výše, do tohoto uložiště jsou data extrahována přímo z primárních systémů. Je zde přesná kopie dat jedna ku jedné. To vše jen proto, aby při následných transformacích nedošlo k nechtěnému poškození původních dat, které by mohlo být nezvratné.

Metody extrakce

Extrakce se může lišit v tom, jestli se snažíme do výstupního uložiště uložit celý obraz vstupních dat nebo se jedná pouze o aktualizaci. V případě aktualizace dat známe dva běžně užívané typy extrakce.

- Přímá extrakce – Jedná se o zachycení nově vzniklých dat hned při vzniku pomocí databázových triggerů, databázových aplikací nebo log souborů.

- Odložená extrakce – V tomto případě se jedná o aktualizaci, která neprobíhá přímo, nýbrž v periodických cyklech, vždy za nějaký časový úsek. Tento způsob probíhá srovnáním zdrojové a cílové databáze např. pomocí časových razítek nebo porovnání souborů.

2.2.2. Transformace

Transformace je proces, který se stará o normalizování a vyčištění dat, získaných extrakcí z primárních systémů. Dostává data do takové formy, aby se shodovala a odpovídala formě datového skladu. (Schiller, 2003)

Data, která získáváme z primárních systémů, jsou často nějakým způsobem nekompletní nebo chybná. Takováto data nazýváme „znečištěná data“ a při procesu transformace se snažíme tyto nečistoty v datech odstranit. Používají se k tomu různé mechanizmy kontroly kvality dat

a také mechanizmy pro jejich čištění. Ve většině případů způsobuje nesoulad a chyby v datech lidský faktor. Může jít například o zaměstnance, kteří mohou chybovat a zadávat špatné údaje nebo to také může být způsobeno neintegrovaností zdrojových systémů. (Schiller, 2003)

Čištění mohou podléhat téměř všechny datové atributy, které nejsou ukazatelového typu. V praxi se často jedná např. o nesprávně zadané adresy zákazníků, nesprávné identifikační údaje, dále pak také transakční atributy jako např. kódy produktů. Transformační procesy musí být schopny chyby a nečistoty detekovat a opravovat. (Vavruška, 2003)

Mezi známé nástroje sloužící k čištění dat patří například software Trillium od firmy Harte-Hanks. Trillium je specializováno na čištění zákaznických dat. Skládá se z modulů, ve kterých dochází k analýze a transformaci dat, dále pak z databází se zeměpisnými údaji, katalogy jmen a tituly. Tyto databáze slouží k opravě chyb v údajích. Podobných nástrojů existuje v praxi celá řada a firmy je hojně využívají. (Vavruška, 2003)

name	address1
Nowmer Sheri	Bailey Road 2433 Tlaxiaco 15057 Mexico
Whelply Derrick	Dewing Avenue Sooke 2219 Canada
Derry Jeanne	7640 First Ave. Issaquah USA 73980
Spence Michael	Tosca Way Burnaby 337 74674 Canada
Gutierrez Maya	8668 Via Neruda Novato USA
Damstra Robert	1619 Stillman Court Lynnwood 90792 USA
Kanagaki Rebecca	D Mt. Hood Circle 2860 Tlaxiaco Mexico

cname	address1	city	postal_code	country
Nowmer Sheri	2433 Bailey Road	Tlaxiaco	15057	Mexico
Whelply Derrick	2219 Dewing Avenue	Sooke	17172	Canada
Derry Jeanne	7640 First Ave.	Issaquah	73980	USA
Spence Michael	337 Tosca Way	Burnaby	74674	Canada
Gutierrez Maya	8668 Via Neruda	Novato	57355	USA
Damstra Robert	1619 Stillman Court	Lynnwood	90792	USA
Kanagaki Rebecca	2860 D Mt. Hood Circle	Tlaxiaco	13343	Mexico

Obrázek 2.3. Rozdíl mezi normalizovanými a nenormalizovanými daty, zdroj: (Lacko, 2003)

Transformace jako taková obsahuje obrovské množství operací, mezi které patří například konverze a filtrování dat, normalizace a denormalizace, vytváření složitých multidimenzionálních struktur, odstranění duplicity nebo matematické operace. (Lacko, 2003)

Cílem transformace je vyčištění a transformování dat. Výsledkem by tedy měla být kvalitní, korektní, očištěná a konsolidovaná data, která mají maximální možnou informační hodnotu a vyhovují formě a struktuře výstupního datového skladu. Eliminují se neshody v názvech položek z různých podnikových zdrojů. Taková data je pak možno vhodně využít pro konstrukci reportů a tvorbu analýz.

2.2.3. Nahrávání

Po provedení transformace jsou transformovaná data konečně nahrána do datového skladu. Mohou na nich být prováděny analýzy nebo můžou být využita v procesu rozhodování. Hlavní problémy, které u nahrávání musíme řešit, jsou především tyto:

- Závislost na cíli – Tím je myšleno na jakých strojích a na jakém hardware se datový sklad nachází. Je třeba umět pracovat s touto platformou a znát specifika pro nahrávání dat do této platformy. (Loshin, 2012)

- Objem dat a frekvence aktualizací – Záleží hlavně na tom, zda bude datový sklad doplňován přírůstkově nebo budou data nahrávána jako výsledek událostí, které spustí příslušné triggery nebo zda data budou nahrávána ve formě periodické aktualizace v určitou dobu (většinou dny, týdny nebo měsíce). (Loshin, 2012)

Z výše uvedeného je patrné, že je nutno mít celý proces nahrávání dobře promyšlen, hlavně co se objemu dat a frekvence aktualizací týká. Stejně tak je nezbytné dobře znát platformu, na které běží datový sklad a umět s ní pracovat. Nejnáročnější částí tvorby je však příprava vhodných datových struktur.

2.2.4. Metadata

Tvorba metadat je velmi důležitou součástí celého ETL procesu a najdeme ji jak v počátečních fázích extrakce, tak ve finálních fázích nahrávání. Proto by bylo dobré se zmínit, co vlastně metadata jsou.

„In order to document the meaning of the data contained in a data warehouse, it is recommended to set up a specific information structure, known as metadata, i.e. data describing data.“ (Loshin, 2012)

Přední odborník v oblasti BI David Loshin (2012) tvrdí, že jsou metadata ve své podstatě data popisující data. Někdy také říkáme „data o datech“. Metadata zaznamenávají informace o originálním zdroji všech dat v datovém skladu. Obsahují jejich význam a také popisují transformace, které byly na datech provedeny. Mohou to ovšem být i jednoduché informace jako například počet sloupců v tabulce apod. Ve spodní části obrázku 2.2. můžeme vidět grafickou prezentaci metadat jako pomyslnou křivku spojující všechny procesy ETL.

Obecně by se dalo říci, že metadata obsahují konceptuální, logickou a fyzickou informaci, potřebnou k transformaci dat z rozdílných množin do souvislé množiny modelů. Zachycují

strukturu dat pro BI, vytvářejí mapu pro realizaci zpětného auditu získávání informací a dávají nám způsob jak sledovat vývoj informací od jejich zisku, přes validaci až po následné reálné využití. (Loshin, 2012)

Je třeba metadata velmi často aktualizovat, aby vždy ukazovala jasný obraz struktury datového skladu a modifikací provedených v této struktuře. Tyto informace by měly být poskytovány všem oprávněným uživatelům skladu.

2.3. ELT

Tento pojem je do jisté míry velmi podobný pojmu ETL nicméně rozdíl je především v přístupu k přesunu dat. Jak je možné si všimnout už ze zkratky ELT (extraction, loading, transformation), pojmy load a transform jsou přehozeny. Celá tato koncepce totiž směřuje k tomu, že se data nejprve nahrají a až potom provádí cílový systém všechny transformace. Jinými slovy, data se zkopírují do datového skladu a až pak transformují. (Speare, 2015)

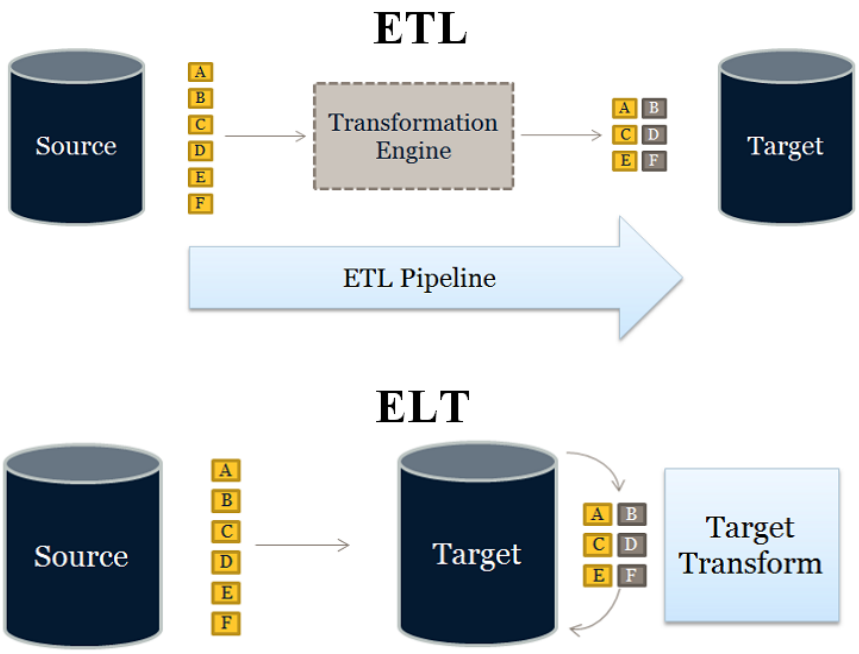
Proč využívat ELT? Celá tato koncepce dává smysl v situacích, kdy je cílový systém vysoce výkonný a není pro něj problematické všechny tyto transformace zvládat. (Speare, 2015)

Většinou se tedy pojí především s technologiemi jako DWA (Data Warehouse Appliance), s Hadoop clustery a s cloudem. Dá se tedy říct, že ELT přístup je nejpoužívanější v souvislosti se zpracováním Big Data. Díky využití transformačního mechanizmu přímo v datovém skladu se dá výrazně zredukovat čas, který stráví data na cestě do skladu, což umožňuje rychlejší realizaci rozhodovacích procesů. Díky tomu je tento způsob ve výsledku o dost efektivnější a ekonomičtější.

2.3.1. Rozdíly ETL a ELT

Není vždy evidentní, který z těchto přístupů je lepší. Každý z nich se hodí pro určité scénáře a situace, které mohou nastat. ELT se řídí mottem „méně je více“, jelikož vlastně postrádá samostatný transformační systém. Data se prostě nahrají a posléze transformují v cíli. (Speare, 2015)

Na druhou stranu ETL funguje spíše na principu potrubí. Data proudí od zdroje k cíli a mezi tím na ně působí transformační mechanizmy, které se starají o změny v datech.



Obrázek 2.4. Srovnání ETL a ELT, zdroj: (Speare, 2015)

Na obrázku 2.4. můžeme vidět schématické vyjádření výše popsaných rozdílů. Je patrné, že u ELT dochází k transformaci dat až v cílové databázi na rozdíl od ETL, kde k transformaci dochází při přesunu do cílové databáze.

Jeden z hlavních důvodů proč se v souvislosti s Big Data ELT v praxi používá, je drastická redukce načítacích časů. Tento fakt nemusí být pravdivý za každé situace. V některých případech může být ETL efektivnější. Ovšem praxe jednoznačně ukazuje, že ve většině případů načítací časy u ELT vedou. To vše díky vysokému a často i distribuovanému výpočetnímu výkonu.

2.4. Datová uložiště

V této části se budeme věnovat typům datových uložišť. Rozebereme jednotlivé varianty od DSA přes ODS, DW až k DM. Budeme se zabývat tím, k čemu se tato uložiště používají a zmíníme podrobnosti o každém z nich.

2.4.1. Dočasné uložiště (DSA)

Jak už název napovídá tak je DSA (Data Staging Area) místem, kde jsou data jen po určitou dobu. Kopírují se zde ze zdrojových systémů. Dočasná uložiště jsou tvořena především

z časových důvodů, jelikož je třeba mít data ze všech DSA připravena už před jejich integrací do datového skladu. (Smith, 2013)

Kvůli různým ekonomickým cyklům, datovým cyklům, limitům sítě a hardware není vždy vhodné extrahovat všechna data ze všech dočasných uložišť ve stejný čas. Nicméně ne pro všechny typy podniků je dočasné uložiště nutnou součástí. V některých případech se vyplatí přesunovat data rovnou z primárních systémů do datového skladu.

2.4.2. Operativní uložiště (ODS)

ODS (Operational Data Storage) neboli operativní uložiště je ve své podstatě velmi podobný pojem jako DSA, nicméně přesto je mezi nimi mnoho odlišností. Jde zde hlavně o rozdíl vnímání ODS a DSA. Například známí odborníci z BI sféry William Inmon a Ralph Kimball mají každý vlastní přístup, který zahrnuje buď ODS, nebo DSA.

Kimball (2013) nahrazuje pomocí DSA ODS. DSA popisuje jako dimenzionální typ uložiště. Tvrdí, že data by po nahrání do DSA neměla být přístupná nikomu jinému než developerům.

Inmon (2005) naopak zavádí pojem operativního uložiště, které má některé podobné vlastnosti jako DSA, ale také se v mnohém liší. Hlavním rozdílem je, že k transformaci dat dochází téměř okamžitě. ODS je chápáno spíše jako uložiště, které je relační, neslouží jen pro dočasné uložení dat, ale má mnohem více využití. Často k němu může být přistupováno i komunitou uživatelů nebo alespoň vybranými členy této komunity. Nicméně obecné vnímání ODS a DSA je velmi podobné.

2.4.3. Datový Sklad (DW)

„The purpose of the Data Warehouse in the overall Data Warehousing Architecture is to integrate corporate data. It contains the single version of truth." (Smith, 2013)

Jak píše Smith (2013), smyslem DW (Data Warehouse) neboli datového skladu je tvořit architekturu pro integraci korporátních dat, s tím že by tato architektura měla obsahovat jedinou verzi pravdy.

Na rozdíl od všeobecně populárního názoru, datové sklady nemusí vždy obsahovat veškerá korporátní data. Jejich hlavním cílem je poskytovat klíčové metriky, které jsou potřebné pro firemní byznys, tvorbu strategií a rozhodování.

Množství dat ve skladu bývá velmi rozsáhlé a úroveň jejich granularity je také vysoká. Mohou například obsahovat data všech prodejů, které kdy firma realizovala a k nim ještě také příslušné dimenze s doplňujícími daty. Tato multidimenzionální datová struktura umožňuje tzv. slicing and dicing datové kostky, což je v podstatě nahlížení na data v různých řezech a souvislostech. Nicméně datový sklad nemusí být nutně multidimenzionální, může být klidně relační. To závisí na způsobu, jakým chce organizace data využívat. Relační typ DW je často označován zkratkou ROLAP (relational online analytical processing) a multidimenzionální typ bývá označován jako MOLAP (multidimensional online analytical processing).

2.4.4. Datové tržiště (DM)

Z jednoho datového skladu může být vytvořeno mnoho datových tržišť. Každé DM je pomocí klasických ETL postupů naplněno daty z datového skladu. Datový obsah daného tržiště je vždy určen podle skupiny tvůrců rozhodnutí, kteří jej využívají. (Smith, 2013)

Datová tržiště mohou být opět relační nebo multidimenzionální podle toho jak budou informace, které obsahují, využívány a podle toho jaké nástroje budeme využívat pro prezentaci těchto dat.

Každé datové tržiště může obsahovat různé kombinace dat z korporátního datového skladu. Příklad využití může být např. v situaci, kdy není třeba rozsáhlých historických dat a hodí se pouze data ze současného kalendářního roku. Personálnímu oddělení se například mohou hodit detailní data o zaměstnancích, ale na druhou stranu datové tržiště, zaměřené na celkový prodej, nepotřebuje mít data jako „plat zaměstnance“ nebo „bydliště“.

Tímto způsobem je vytvořen nespočet datových tržišť, nad kterými pak pracují nástroje pro analýzy, reporting nebo pro vizualizaci dat. Nevýhodou je, že když jsou tržiště tvořena dříve než datové sklady, je při jejich slučování nutno znovu opakovat transformační proces.

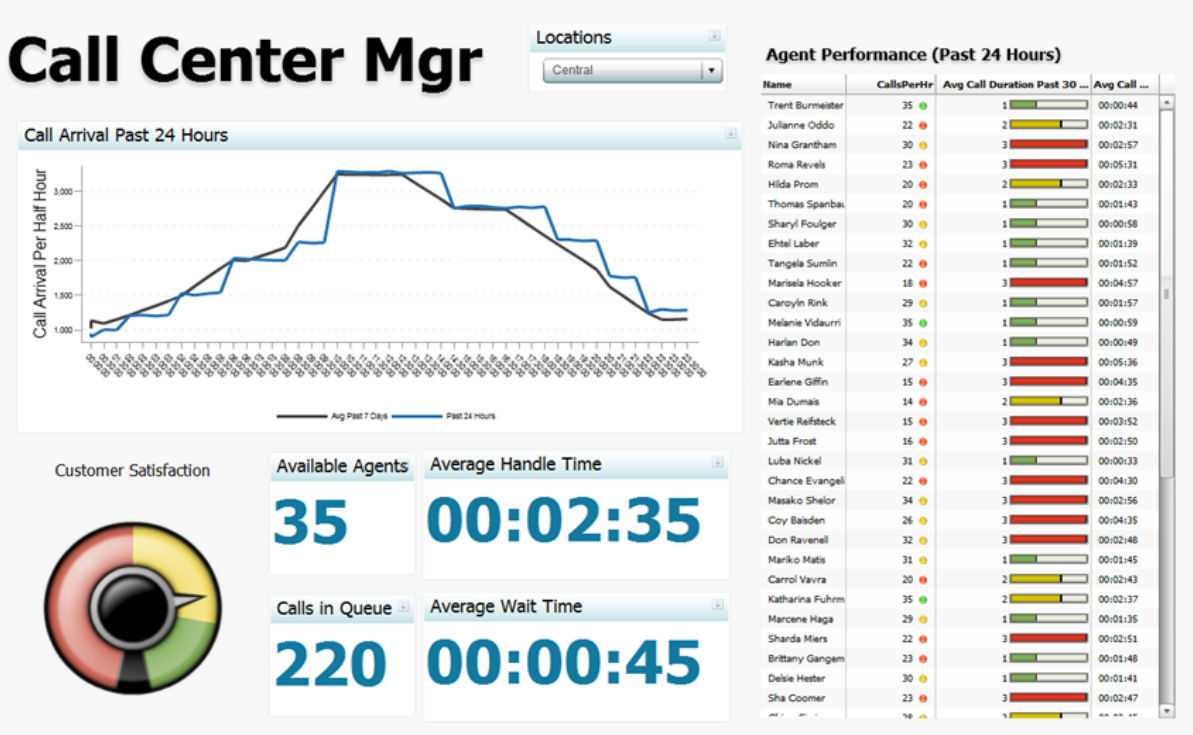
2.5. Reporting a vizualizace dat

Reporting a vizualizace dat patří, stejně jako ETL, mezi esenciální prvky business intelligence. Díky těmto metodám je možno vyobrazit data nejen pomocí nepřehledných tabulek, nýbrž pomocí grafů, map, křivek a spousty jiných grafických reprezentací.

Slouží tedy k jednoduššímu pochopení dat a podnikání jako celku. Vede k přehlednějšímu a intuitivnějšímu vnímání souvislostí. Některé průzkumy ukazují, že zavedení vizualizace dat

do firemního reportingu může zlepšit chápání firemního podnikání až o 74%, což naznačuje důležitost této oblasti. (Eckerson, 2011)

Nejedná se nikdy o univerzální řešení, jelikož každé odvětví a každá společnost potřebují specifický typ reportů a dashboardů. Jejich vzhled se tedy může velmi lišit. Například reporty, které potřebují sledovat běžní pracovníci, se mohou lišit od reportů, které sleduje vedení. Ukázku dashboardu můžeme vidět na obrázku 2.5. (Eckerson, 2011)



Obrázek 2.5. Dashboard manažera call centra, zdroj: (Aanderud, 2012)

Reporting a vizualizace dat jsou dnes velmi kýženými artikly v oblast BI. Velké množství firem se tak snaží přejít k používání interaktivních a velmi často i dynamických reportingových nástrojů, které jim umožní sledovat jejich podnikání takřka nepřetržitě. Začínají si totiž uvědomovat, že v grafické prezentaci je síla, která umožní jednoduché pochopení složitých souvislostí. Grafy a dashboardy se tedy stávají běžnými součástmi podnikových prezentací.

Rozmach celé této oblasti také do jisté míry ovlivňuje rozmach internetu a virtualizace. Vysoká dostupnost připojení ve světě umožňuje tvorbu serverových aplikací, jež neustále nabízí vizualizace aktuálních podnikových dat. Tyto vizualizace může vrcholový management sledovat téměř kdykoliv a odkudkoliv. Tím management získává neustálý přehled o výkonnosti podniku a může podstatně rychleji reagovat na problémy, které by mohly způsobit pokles výkonu.

Je tedy patrné, že vizualizace dat může být (pokud je správně využita) velmi silným nástrojem, obzvláště pro podnikové vedení. Toto se týká podniků z velice různorodých oblastí například zdravotnictví, průmysl, ekonomická sféra, informační technologie a mnoho dalších.

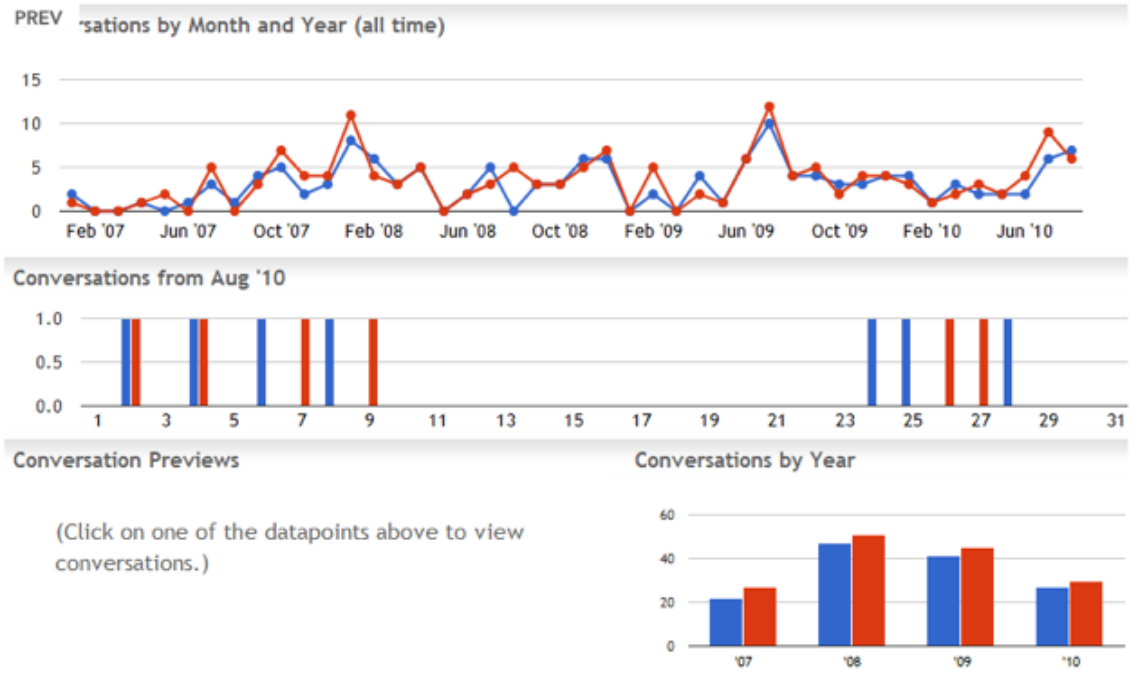
2.5.1. Vývoj vizualizace dat

Tvrdí se, že první koláčové, sloupcové a spojnicové grafy byly použity už okolo roku 1700 skotským inženýrem a politickým ekonomem Williamem Playfairem. Playfair graficky vyobrazil růst Britského národního dluhu mezi lety 1688 a 1800 v grafu zvaném „Chart of the National Debt of England“. Proto je oficiálně považován za zakladatele grafické reprezentace dat. (Eckerson, 2011)

Grafická reprezentace dat se během posledních dvou století stala používaným nástrojem především v přírodních vědách nebo v ekonomii, nicméně koncem 20. století se začala postupně dostávat i do korporátní a zákaznické sféry.

S rozmachem internetu se stala vizualizace velmi kýženou. Jak už bylo řečeno výše, firmy se snaží tvořit přehledné reporty a dashboardy pro vedoucí pracovníky, aby usnadnily rozhodování a tvorbu všeobecného přehledu o chodu firmy.

V zákaznické sféře se jedná především o trendy posledních let. Například Google vydal v roce 2010 uživatelský nástroj zvaný Graph Your Inbox, který slouží k zobrazení grafických přehledů emailové schránky uživatele. Uživatel může například sledovat emailovou aktivitu nebo trendy. Výstup vizualizace uživatelských dat GYI můžeme vidět v obrázku 2.6. Dále můžeme zmínit produkt firmy Apple zvaný Apple Watch. Jedná se o hodinky, jež neustále monitorují tělesnou aktivitu uživatele. Srdeční tep, množství kalorií, které uživatel při pohybu spaluje nebo jaký počet kilometrů urazil. Svá osobní data může uživatel sledovat v grafech, které mu pomohou nastavit si ideální pohybový režim, pokud chce např. zhubnout.



Obrázek 2.6. Přehled služby Graph Your Inbox, zdroj: vlastní zpracování

2.5.2. Vizualizace dat v podnikové sféře

Grafické reprezentace dat vysvětlí vzory, trendy a souvislosti mnohem rychleji než tabulky a text. Díky vizualizacím mohou podnikoví uživatelé zaregistrovat problémy, které je zdlouhavé rozpoznat z tabulek, téměř ihned a mohou rychle učinit příslušnou reakci. V textových reportech a formulářích zůstanou trendy a problémy mnohem déle ukryty v obrovském množství polí, čísel a textu. Díky své vysvětlovací a komunikační schopnosti se tak vizualizace dat stává nedílnou součástí podnikové ekonomiky. (Eckerson, 2011)

"With the graphics, it's much easier for the eye and the brain to understand what's happening in the data. It helps people see what is going to happen rather than what has already happened, and I think that's a natural progression of analytics." (Eckerson, 2011)

Ovšem i v dnešní době je obrovské množství firem, které (ke své vlastní škodě) vizualizaci dat nepoužívají a stále se „brodí" nekonečnou haldou nepřehledných tabulek a formulářů. Tento problém bývá způsoben starými zajetými zvyklostmi managementu. Jak říká známé české přísloví „starého psa novým kouskům nenaučíš". Tímto příslovím by se trefně dal shrnout přístup vedoucích pracovníků ve spoustě firem, jež jsou léta zvyklí na procházení rozpočtů a plánů v tabulkových souborech Excelu. Stejně tak podnikoví analytici, kteří příliš lpí na Excelu, mohou být občas skeptičtí k vizualizaci dat. (Eckerson, 2011)

Nicméně poptávka po vizualizaci dat neustále narůstá. Obzvláště díky přechodu od klasických statických grafů a reportů k interaktivním, dynamickým variantám.

2.5.3. Výhody vizualizace dat

Obecně jsme se o výhodách datové vizualizace zmínili již v předchozí kapitole. Ve zkratce se tedy jedná o rychlejší a intuitivnější pochopení dat, které umožní rychlou reakci. Nyní však výhody shrneme v bodech.

Porozumění podnikání

Podle průzkumů podniků, jež provedl Eckerson (2011), necelé tři čtvrtiny zaměstnanců tvrdily, že vliv vizualizace dat na pochopení podnikání jejich firmy je velmi vysoký. Tento fakt také dokazuje na firmě Advance Auto Parts, která po zveřejnění nového výkonnostního dashboardu provedla měření počtu zobrazení tohoto dashboardu za den. Dashboard byl přístupný tisícům zaměstnanců a počet zobrazení byl mnohonásobně větší než u ostatních firemních informačních systémů. Firma proto provedla průzkum mezi zaměstnanci, ze kterého vyplynulo, že dashboard je u zaměstnanců na suverénním prvním místě v žebříčku firemních informačních prostředků.

Efektivnější způsoby řešení

Vizualizace pomáhá vnímat práci jednotlivců odlišně. Analytická kultura firmy se tak mění a zaměstnanci mají „otevřené oči“. Díky grafické reprezentaci vidí nové způsoby řešení, které by je předtím nenapadly. Mohou pak tedy dělat svou práci lépe a efektivněji, což pomáhá vylepšit strategii celé firmy.

Vliv na poptávku po BI řešení

Mnoho firem si neuvědomuje výhody vizualizace dat až do doby, kdy ji začne využívat. Když firma objeví důležitost vizualizace dat, často investuje do vizualizace a kompletního BI řešení mnohem větší množství peněz. BI řešení je pak možno mnohem snadněji prodat, což přináší zisky pro vývojáře BI a reportingových nástrojů.

2.5.4. Technologie pro vizualizaci dat

Jsou zde dvě hlavní kategorie, do kterých se technologie pro vizualizaci dat dělí a sice: vizuální reporting a vizuální analýza, takže si tyto kategorie přiblížíme. Dále také definujeme pojmy metrika a ukazatel, které bývají často zaměňovány a vnímány velmi nesourodě.

Vizuální reporting

Vizuální reporting se soustředí především na tvorbu grafů a grafických prostředků k zachycení výkonu podniku. Činí tak pomocí vykreslování časových řad a metrik. Základním typem vizuálního reportu je tzv. dashboard, který dává uživateli přehled firemního výkonu. Nejkvalitnější dashboardy mají funkci nazývanou drill down. Ta umožňuje dostat se do další úrovně pro zobrazení detailnějších informací o dané metrice. Dashboard by tedy měl sloužit především k zvýraznění výkonnostních anomálií pomocí vizualizačních technik. (Eckerson, 2011)

Vizuální analýza

Na druhou stranu vizuální analýza umožní uživatelům zkoumat data za účelem objevení nových poznatků a vztahů. Zatímco vizuální reporting staví navigaci v datech na předdefinovaných metrikách, vizuální analýza se snaží poskytnout uživateli mnohem větší úroveň datové interaktivity. Uživatelé mohou vizuálně srovnávat, filtrovat nebo korelovat data během několika sekund. Nástroje vizuální analýzy také často zastřešují i tvorbu predikcí, modelování a statistických what - if analýz. (Eckerson, 2011)

Metrika

Metriky jsou také někdy chápány jako fakta. Jedná se o číselné údaje, o přesné měřitelné hodnoty. Například cena objednávky nebo celkový zisk firmy může být metrikou. Metriky jsou při reportingu podle potřeby sumarizovány a průměrovány popř. rozdělovány na části pomocí ukazatelů.

Ukazatel

Ukazatel můžeme chápat jako údaj, který nám rozděluje daná metrická data do dimenzí. Jedná se o popisný údaj. Nejde o konkrétní číselné hodnoty, ale údaje, podle kterých rozdělíme hodnoty metrik. Když budeme například brát celkový zisk firmy v korunách jako metriku. Jednotlivé firemní divize budou ukazatele, které nám rozdělí metriku firemního zisku na několik částí podle divize. Dále může být ukazatel např. číslo prodejce nebo jméno zákazníka.

2.5.5. Typy vizualizací dat

V této části se budeme věnovat různým typům datových vizualizací, jež se v praxi používají. Spousta z nich bude využita v rámci realizace praktické části práce, takže se alespoň seznámíme s některými základními typy vizualizací, které budeme používat.

Sloupcový graf

Jedná se o jeden ze základních způsobů jak graficky vyobrazit nezávislé veličiny. Typicky se používají pro vyobrazení různých množství dat s jednou proměnnou. Sloupcový graf bývá orientován jak horizontálně, tak vertikálně dle potřeby. Je možné tímto grafem vyobrazit i více veličin a utvořit tak jakési skupinky sloupců. Například sledovat vždy 3 různé ukazatele za časové období. (Laumans, 2009)

Skládaný sloupcový graf

Tento typ sloupcového grafu je primárně určen pro znázornění vztahu jednotlivých položek k celku, s porovnáním podílu každé z hodnot na celkové hodnotě. To vše ještě v různých kategoriích. Stejně jako klasický sloupcový graf se často vyskytuje jak ve vertikální, tak v horizontální formě. Ideální pro vyobrazení např. podílu jednotlivých pracovišť na celkovém zisku za období a podobně.

Spojnicový graf

Většinou se pojí s vyjádřením trendu vývoje v čase. Tzn. na horizontální ose grafu bývá ve většině případů časový údaj. Můžeme takto sledovat vývoj několika různých časových řad. Jde tedy o kontinuální časové řady. Tento typ vizualizace je velmi často využíván pro sledování vývoje důležitých firemních metrik v čase.

Plošný graf

Jedná se opět o graf vyjadřující trend, nicméně tentokrát zde můžeme díky dvojrozměrnému provedení rozložit tento trend na podsložky a identifikovat velikost vlivu jednotlivých podsložek pomocí plochy, kterou pokrývají. Princip je podobný jako u skládaného sloupcového grafu. Můžeme například vidět vývoj firemního příjmu, kde každá rozdělená oblast bude představovat jednu z oblastí příjmu.

Koláčový graf

Koláčový graf slouží k vizuálnímu srovnání proporčních dat. Může proporce vyjadřovat jak počtem pozorování, tak procentuálním poměrem k celku. V praxi se s tímto typem grafu setkáme například při srovnání volebních preferencí. Tyto grafy jsou efektivní jen při srovnávání několika typů možností s hojným zastoupením u každé z nich. Při obrovském množství možností s malým zastoupením se tento graf stává nepoužitelným.

Prstencový graf

Prstencový graf je složitějším typem koláčového grafu, který slouží k porovnání proporcí dvou a více veličin. Vnější prstenec je první z veličin a vnitřní druhá. V praxi může být nahrazen několika koláčovými grafy vedle sebe, ale k ušetření místa v prezentacích a dashboardech se často používá prstencová varianta. (Laumans, 2009)

XY bodový graf

Tato varianta bodového grafu se používá obzvláště pro vizualizaci korelací mezi proměnnými. Pro analytiky velmi známý graf, který se běžně vyskytuje pod zkratkou SP (Scatter Plot). Pomocí SP se hledají závislosti mezi daty, které nejsou na první pohled viditelné z tabulek. Je možné ho použít jen dvojrozměrně. (Laumans, 2009)

Bublinový graf

Jedná se opět o mutaci bodového grafu, která ale na rozdíl od SP přidává možnost vícerozměrných dat. Podobně jako jsou v SP tvořeny body, tak zde jsou tvořeny bubliny a jejich velikost indikuje hloubku na ose z. Takže různě velké bubliny mají různou z souřadnici v souřadnicovém systému. Používá se tedy k dvojrozměrnému vyjádření vícerozměrných dat. (Laumans, 2009)

Stromový diagram

Je velmi často používán k reprezentaci striktních datových hierarchií. Takže se používá k reprezentaci rodokmenů nebo k reprezentaci databázových hierarchií a struktur. Založen na klasickém přístupu: rodič-dítě, předek-potomek. (Laumans, 2009)

Geografická mapa

Mapy jsou jedním z trendů vizualizace dat v posledních letech. Jedná se o využití reálných geografických map pro vyobrazení dat o různých oblastech. Například pokud je firma nadnárodní a má pobočky v různých státech, může vizualizovat data v rámci jednotlivých států. Stejně tak to může fungovat i s výsledky v rámci měst. Většinou jde o kombinaci map s grafy a drill down možnostmi. (Laumans, 2009)

3. Výběr a analýza Open Source Business Intelligence nástrojů

V první části této kapitoly se seznámíme s projektem, který je zadáním praktické části diplomové práce, jeho specifikacemi, podmínkami atd. Dále analyzujeme a shrneme současné řešení daného problému ve firmě. Poté se však budeme věnovat nejdůležitější části této kapitoly, což je analýza Open Source BI nástrojů, které byly vybrány jako vhodní kandidáti pro realizaci řešení projektu.

3.1. Dodavatel řešení

Projekt byl realizován společností Tieto, která je jednou z největších korporací v oblasti IT služeb nejen v České republice, ale i v celé Evropě (obzvláště pak severní části). Firma se soustřeďuje téměř výhradně na severoevropský trh. Největší množství klientů má ve svých domovských zemích, což jsou Finsko a Švédsko. Klienti pochází jak z veřejného, tak ze soukromého sektoru, např. velké množství severských vládních institucí má svá IT řešení realizována firmou Tieto.

Firma je gigantickým dodavatelem IT služeb všeho druhu a tvorba BI řešení patří také do jejího portfolia. Dále však do její nabídky patří: podnikové systémy, integrování obchodních procesů, vývoj a správa aplikací, nabídky konzultace či poradenství v oblasti IT a v neposlední řadě také výzkum a testování nových technologií.

Hlavní centrála Tieto pro ČR je v Ostravě a zaměstnává dnes už něco přes 2000 zaměstnanců. Nedávno byla otevřena nová pobočka v Brně, která je však zatím velmi malá. Díky nově získané zakázce pro Švédsko plánuje ostravská pobočka Tieto najmout během roku 2016 až 400 nových zaměstnanců.

3.2. Zadání projektu

Zadavatelem projektu je firma, která je dlouholetým zákazníkem Tieto, avšak z důvodu práce s citlivými firemními daty v projektu a firemní politice si nepřeje být jmenována. V práci tedy nebudou zveřejňována žádná konkrétní data, nýbrž jen postupy tvorby řešení. Všechny tabulky, grafiky nebo přílohy obsahující konkrétní data budou znečitelněna.

V projektu se jedná o realizaci dynamického reportingového řešení, které by mělo reportovat data získaná pomocí software Jira a to data týkající se chyb a chybových hlášení na serverech zákazníka.

Zákazník si přál řešení pomocí vytvoření databáze na jednom z Tieto serverů, do které budou periodicky nahrávána data získaná z Jiry. Dále by měl být zvolen SW sloužící k reportingu a vizualizaci těchto dat, který neustále bude pracovat s touto databází, bude vždy přístupný a bude poskytovat managementu nejnovější verzi dat.

Důležitou podmínkou je, že zákazník si nepřeje používat žádné drahé nástroje. Jde mu o to, aby bylo celé řešení pokud možno co nejlevnější. Je tedy třeba se vyhnout nástrojům s drahou licencí. Funkcionality, které zákazník požaduje, nejsou až tak rozsáhlé a proto bude SW nástroj vybrán z oblasti Open Source (nástroje s bezplatnou licencí) BI nástrojů.

Zadání obsahuje ještě drobné podmínky například že, výsledné grafy a vizualizace by měly dodržovat firemní koncept, styl a barevné spektrum, které bylo pro tyto vizualizace používáno doposud.

3.3. Analýza dosavadního řešení

Reportingem serverových chyb typu bug (chyby v software) pro tohoto klienta se Tieto už nějakou dobu zabývá. Doposud však byl celý proces velmi neefektivní. V této části se zmíníme o některých problémových oblastech stávajícího řešení.

3.3.1. Problémy s databázovým přístupem

Kvůli složité firemní politice zákazníka je problematické získat přímý přístup do jejich databází. Je to paradoxní, ale bylo by zapotřebí mnoho papírování a povolení z různých pracovišť zákazníka. Proto jsou prozatím data získávána exportem do několika jednoduchých datových souborů (Flat file) typu XLSX (soubor Microsoft Excel), které jsou tvořeny jednou měsíčně.

Tato situace se bohužel nejspíše v brzké době nezmění a nezbývá nám, než při řešení získávat data stejným způsobem a následně provádět extrakci, transformaci a nahrávání do námi vytvořené databáze na serveru Tieto.

3.3.2. Neefektivní řešení pomocí MS Power Pivot

Doposud celý reporting probíhal tak, že analytik, který se tímto projektem zabývá, musí každý měsíc ručně dělat export dat. Poté vytvářet kontingenční tabulky a následně opakovaně tvořit ty stejné grafy pro každý měsíc. Tato zdlouhavá a časově náročná práce mu vždy na konci měsíce zbytečně zabere téměř dva pracovní dny.

Dle výše uvedeného je dosavadní řešení vysoce neefektivní, zbytečně pracné a časově náročné. Mohlo by být mnohem lépe automatizováno, to znamená velké množství získaného času, který by analytik mohl využít podstatně efektivněji.

3.3.3. Zhodnocení dosavadního řešení

Ukázalo se, že další nevýhodou je nepřístupnost daného řešení online. Grafy a vizualizace vytvořené v MS Power Pivot jsou každý měsíc prezentovány pro management zákazníka a jinak jsou zcela nepřístupné.

Cílem realizace projektu je tedy všechny tyto problémy eliminovat a vytvořit efektivní, levné a přístupné řešení reportingu pro zákazníka. Dosavadní řešení vůbec nebralo v úvahu možnost online přístupu a automatizace celého procesu reportingu.

A proto je zde snaha vyřešit tento problém pomocí dynamického reportingu, kde budou data v databázi a budou v reálném čase vizualizována. Analytikovi pak odpadne spousta hodin zbytečné práce, jelikož grafy budou tvořeny dynamicky a jejich šablony automaticky načtou nová data při každém obnovení.

3.4. Výběr OS nástrojů vhodných pro řešení

Po zhodnocení dosavadního řešení a uvážení všech požadavků zadavatele jsme se rozhodli realizovat řešení pomocí Open Source BI nástrojů. V této kapitole bude vybráno a popsáno několik nástrojů, které by mohly být pro naše řešení vhodné. Zmíníme se o jejich základních funkcionalitách, co všechno lze pomocí nich řešit, případně jaká omezení mají. V oblasti OS software se dá najít mnoho solidních nástrojů, ovšem základním problémem je nedostatek podpory a celková pracnost při použití těchto řešení.

Po dlouhodobějším bádání na poli OS BI nástrojů zaměřených převážně na reporting a vizualizaci dat, jsme vybrali ty, které jsou potenciálně vhodné pro řešení našeho problému. Řídili jsme se přitom pokyny, které plynou z požadavků zadavatele projektu.

Nakonec jsme vybrali sedm nástrojů do širšího výběru, který podrobíme zhodnocení a analýze. Názvy nástrojů můžete vidět abecedně seřazené v níže uvedeném seznamu.

Seznam vybraných OS BI nástrojů:

- Birt,
- JasperReports,
- Jedox Base,
- Pentaho,
- Seal Reports,
- SpagoBI.

Všechny nástroje postupně v následujících kapitolách projdeme a pokusíme se zúžit výběr na tři výsledné nástroje podle toho, zda jsou vhodné pro splnění zadání projektu. Z výsledných tří nástrojů poté pomocí dotazníku podaného třiceti BI specialistům z Tieto vybereme finální a nejvhodnější nástroj, pomocí kterého budeme projekt realizovat.

Základní informace a všech výše zmíněných nástrojích a jejich zhodnocení, obsahují následující podkapitoly.

3.4.1. Birt

„BIRT is an open source software project that provides the BIRT technology platform to create data visualizations and reports that can be embedded into rich client and web applications, especially those based on Java and Java EE. BIRT is a top-level software project within the Eclipse Foundation, an independent not-for-profit consortium of software industry vendors and an open source community.“ (The Eclipse Foundation, 2014)

Jak můžeme vidět už z výše uvedeného popisu programu, který je převzat z oficiálního webu Birt, jedná se o specializovaný nástroj pro tvorbu datových vizualizací, který je celý postaven na programovacím jazyku Java. Je výsledkem projektu Eclipse Foundation, což je firma, která se stará o vývoj známého programátorského nástroje Eclipse.

Software je koncipován jako reportingový nástroj, dohromady s runtime komponentou, která je uzpůsobena na deploy (umístění) do jakéhokoliv Java prostředí. Typ licence je open source.

Skládá se z několika nástrojů, které je možno stáhnout buď samostatně, nebo v kompletním balíčku, který obsahuje všechny nástroje.

Birt tedy obsahuje tyto nástroje:

- Birt Report Designer – Nástroj pro návrh reportů. Stažitelný buď jako rozšíření Eclipse, nebo jako kompletní balík Eclipse a Report Designer v jednom.
- Report Engine – Slouží ke generování a renderování reportů. Může být vložen v jakékoliv Java EE aplikaci, což ho činí velmi versatilním.
- Charting Engine – Slouží k tvorbě grafů, které mohou být opět samostatné nebo vložené v jakékoliv Java EE aplikaci.
- Birt Viewer – Nástroj sloužící k prohlížení náhledů na reporty přímo v Eclipse. Má v sobě integrovaný Apache Tomcat server a může tudíž fungovat v server režimu. Náhled je možno vytvořit ve formátech HTML, PDF, XLS, DOC, PPT a také exportu do CSV.

Zhodnocení Birt

Birt je poměrně zdařilý software, který může uspokojit většinu reportingových potřeb.

Mezi jeho výhody patří:

- Rozsáhlé reportingové možnosti – grafy, reporty, vizualizace,
- Zvládá jakékoliv množství dat,
- Bezproblémové navázání konektivity s většinou běžných typů databází,
- Vcelku intuitivní ovládání,
- Nedělí se na OS komunitní a placenou verzi, existuje jediná, která je zcela zdarma.

Mezi nevýhody patří:

- Nutnost instalovat každý dílčí nástroj zvlášť,
- Problémy s podporou ovladače JDBC databáze, který je pro naši práci stěžejní,
- Dlouhé trvání vykreslení reportů (obzvláště tabulek),
- Nemoderně působící vizualizace, ač jsou možnosti vizualizace rozsáhlé, některé typy grafů či KPI indikátorů působí fádně a nevzhledně.

3.4.2. JasperReports

„The JasperReports Library is the world's most popular open source reporting engine. It is entirely written in Java and it is able to use data coming from any kind of data source and produce pixel-perfect documents that can be viewed, printed or exported in a variety of document formats including HTML, PDF, Excel, OpenOffice and Word.“ (Tibco Software, 2016)

Jak tedy můžeme vidět výše, opět se jedná o OS reportingový software, který je celý napsán v jazyce Java. Tento software je dílem společnosti Tibco Software, což je firma zabývající se vývojem business intelligence aplikací pro podnikovou sféru, zaměřených především na rychlý zisk dat a vizualizace. Stejně jako u Birt jde o rozšíření vývojářského nástroje Eclipse.

Jedná se o koncept ETL nástroje v kombinaci s reportingovým nástrojem a instancí serveru, který se skládá z několika částí.

Části JasperReports jsou:

- JasperReports Server – server, který může fungovat buď samostatně, nebo může být vložen v mobilní či webové aplikaci. Je optimalizován pro sdílení a správu Jasper reportů.
- Jaspersoft ETL – klasický ETL nástroj.
- Jaspersoft Studio – Slouží k tvorbě reportů, které lze buď exportovat do velkého množství formátů (HTML, PDF, XLS aj.) využitelného pro prezentace, nebo deploy (umístění na server).

V případě JasperReports se už nejedná o 100% Open Source. Společnost nabízí software ve třech různých verzích licence. Jedná se o verze Community, Commercial for internal business use a Commercial for applications.

Komunitní verze je tedy OS, zatímco komerční verze jsou placené buď pro vnitřní využití v rámci firmy, nebo pro využití jako součástí jiné komerční aplikace třetí strany. V následujícím obrázku můžeme vidět srovnání obsahu všech tří verzí.

	Community	Commercial	Commercial
Licensing available for:	Business use and open source apps	Internal business use	Commercial applications
1. Free Access: Do you prefer to download and use publicly available software binaries or source code at no cost?	•		
2. Embeddability and Distribution Rights: Do you want to OEM, include, or embed Jaspersoft software in a publicly distributed product, application, appliance or service?	Under terms of GNU GPL license		JASPERSOFT Under terms of agreement
3. Certified Platform Support: Do you use a proprietary operating system, application server, or database?		•	•
4. Managed Release Cycles: Do you want product enhancements rolled up into fewer releases?		•	•
5. Support Guarantees: Do you need guaranteed support for mission-critical and production applications, or support for older releases?		•	•
6. Legal Matters: Do you require product warranties and indemnification?		•	•
7. Advanced Functionality: Do you need value-added features, such as Web-based ad hoc query, in-memory analysis & reporting?		•	•

Obrázek 3. 1. Srovnání balíčků JasperReports, zdroj: (Tibco Software, 2016)

Zhodnocení JasperReports

Mezi výhody JasperReports patří:

- Rozsáhlé možnosti reportingu a vzhledné vizualizace dat,
- Velké množství kvalitních předpřipravených šablon,
- Zvládá jakékoliv množství dat,
- Podpora velkého množství databázových ovladačů.

Nevýhody JasperReports jsou:

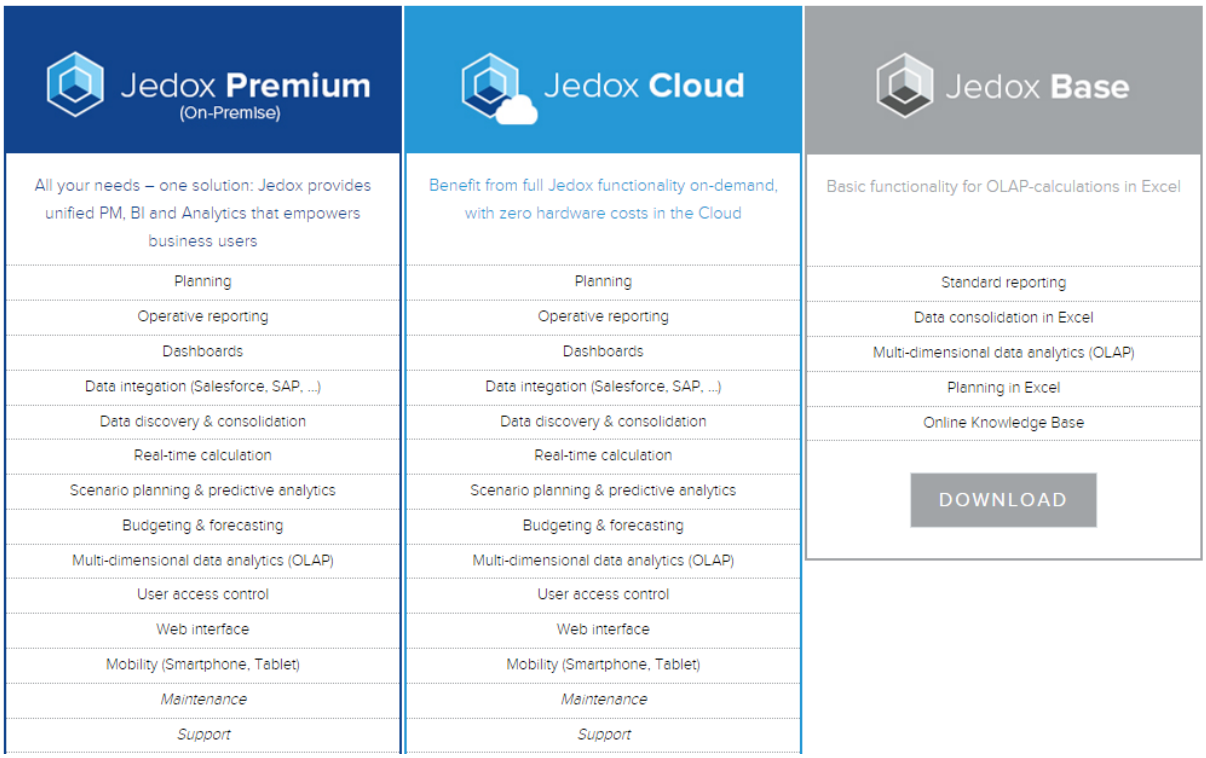
- Slabý ETL nástroj – existují mnohem lepší OS ETL nástroje od jiných firem, které fungují o poznání lépe, jsou přehlednější a uživatelsky přívětivější,
- Nevýhoda placených licencí – OS veze je osekaná o některé důležité prvky,
- Neúplná funkčnost některých nabízených možností.

3.4.3. Jedox Base

„Jedox gives you straightforward self-service BI, empowering you to make faster decisions. Jedox optimizes any business processes with a unified solution that you can use from Excel, on the web, on tablets and smartphones, and in the cloud. Jedox drives growth, innovation, and collaboration by bringing the power of data to every user in your organization.“ (Jedox AG, 2016)

Jedox Base je nástroj firmy Jedox AG, který je postavený na tom, že se jedná o rozšíření MS Excelu (podobně jako Power Pivot). Ovšem na rozdíl od Power Pivot má Jedox base i svou vlastní serverovou instanci a je tedy přizpůsoben k online reportingu.

Je poskytována ve třech licenčních variantách. Jedox Base je základní OS licence, která je zdarma. Dále pak nabízí i cloudové řešení v podobě Jedox Cloud. Posledním typem licence je Jedox Premium, která představuje kompletní řešení se vším všudy. Srovnání obsahu jednotlivých licenčních balíčků můžeme vidět na obrázku 3.2.



Obrázek 3. 2. Srovnání balíčků Jedox, zdroj: (Jedox AG, 2016)

Zhodnocení Jedox Base

Jedox Base je zajímavý přístup k reportingovému řešení. Jako problematická se však jeví jeho přímá závislost na MS Excel. Bez zakoupení licence na MS Excel je tedy absolutně nevyužitelný.

Mezi výhody Jedox Base patří:

- Opět kvalitní a dobře vypadající vizualizace,
- Zvládá jakékoliv množství dat,
- Konsolidace dat v Excelu,
- Nadstavba Excelu – zjednoduší práci spoustě uživatelů.

Nevýhodami Jedox Base jsou:

- Závislost na MS Excel – jelikož Excel není open source, nutí to zákazníka zaplatit si licenci Excelu,
- Cloudové řešení, které by mohlo být velmi efektivní, je až v placené verzi,
- Základní verze neobsahuje ETL nástroj,
- OS verzi chybí některé ze stěžejních prvků,
- Problémy s některými možnostmi a jejich funkčností.

3.4.4. Pentaho

„Pentaho's strong open source heritage drives an extensible, pluggable and open platform with flexibility for a wide range of visualization and analytic options, from reporting to predictive analytics. Pentaho delivers the greatest value for a complete enterprise-class business analytics solution fueled by an open innovation model." (Pentaho Corporation, 2016)

Pentaho nabízí také řešení pomocí koncepce ETL nástroje, reportingového nástroje a serverové instance. Jedná se tedy opět o klasickou kombinaci uzpůsobenou pro online reporting a prediktivní analýzy. Nicméně nabízí toho mnohem více, jelikož ve své placené verzi nabízí široké možnosti pro zpracování Big Data a mnoho dalšího.

Kromě Pentaho Community verze, jež je OS a zcela zdarma, má Pentaho také Enterprise Edition, která nabízí velmi rozsáhlé možnosti, ale je placená. Srovnání nabízených možností obou licencí můžeme vidět v příloze č.1.

Bi řešení od Pentaho se tedy skládá z následujících částí:

- Bussiness Analytics Platform – Umožňuje podnikovým uživatelům objevovat a promíchat všechna data. Velké spektrum analytických nástrojů: reporting, prediktivní modelování, uživatelé mohou analyzovat a vizualizovat data napříč různými rozměry a dimenzemi.
- Data Integration – (pod názvem Kettle) nabízí výkonné ETL funkcionality. Můžete použít tuto samostatnou aplikaci k vizuálnímu a logickému návrhu úloh pro extrakci a přípravu dat.
- Report Designer – Slouží k tvorbě reportů z dat bez nutnosti jakýchkoliv zprostředkujících tabulek. Podporuje export do PDF, XLS, HTML, XML a CSV.

- Marketplace – Správa pluginů a testování pluginů, které mohou rozšířit možnosti Pentaho.
- Aggregation Designer - poskytuje jednoduché rozhraní, které umožňuje vytvářet deployovat souhrnné tabulky pro zlepšení výkonu OLAP kostek.
- Schema Workbench - je vizuální návrh rozhraní, které umožňuje vytvářet a testovat Mondrian OLAP kostky.
- Metadata Editor - nástroj, který zjednodušuje tvoření dat o datech, umožňuje vytvořit metadatové domény a relační datové modely.
- Hadoop Shims - používá abstraktní vrstvu pro usnadnění podpory pro velkou škálu Hadoop distribucí pro práci s Big Data. Softwarové moduly rozšiřující funkce aplikace (pluginy), které umožňují kompatibilitu s konkrétní distribucí, se nazývají „Shims“.

Zhodnocení Pentaho

Firma Pentaho je poměrně známým hráčem, který má velké množství zákazníků. Nicméně toto se týká Enterprise verze jejich produktu. OS verze je velmi ořezaná a většinu nástrojů buď neobsahuje, nebo jsou velmi omezené. Přesto patří k tomu nejlepšímu, co se dá v oblasti OS business intelligence řešení použít.

Výhody Pentaho Community jsou:

- Rozsáhlé možnosti reportingových nástrojů reporty, charty atd.,
- Zvládá jakékoliv množství dat,
- Konektivita s obrovským množstvím DTB bezproblémová,
- Velmi zdařilý a přehledný ETL nástroj,
- Velmi intuitivní ovládání.

Nevýhody Pentaho Community jsou:

- Omezené možnosti většiny nástrojů, obzvláště pro práci s big data,
- Nedostatek předpřipravených šablon.

3.4.5. Seal Reports

„Seal Report offers a complete framework for producing every day reports and dashboards from any database. The product focuses on an easy installation and report design:

Once setup, reports can be built and published in a minute. Seal Report is an Open Source for the Microsoft .Net Framework entirely written in C#." (Ariacom, 2016)

Jde o projekt firmy Ariacom, který nabízí 100% Open Source reportingové řešení. Soustředí se především na reporting a má koncepci serverové instance v kombinaci s reportingovým nástrojem. Je celý napsán v programovacím jazyce C# (see sharp).

Skládá se z těchto částí:

- Web Report Server – server sloužící k online přístupu k datům a prezentaci výsledných reportů.
- Report Engine – nástroj sloužící k tvorbě reportů, grafů a dashboardů, které lze také vyexportovat do formátů HTML, XLS, PDF, DOC a také do formátu CSV.

Zhodnocení Seal Reports

Seal Reports se ve výsledku jeví jako velmi těžko použitelný nástroj. Chybí mu jakékoliv grafické rozhraní a celý vzhled reportu je nutno kódovat pomocí CSS souboru. Popř. pomocí použití předloh grafů ve formátu NVD3. Absence grafického rozhraní takřka znemožňuje tvorbu reportů pracovníkům z vedení, kteří se nevyznají v CSS kódování. Proto mezi testovanými nástroji patří tento nástroj k těm horším.

Mezi výhody Seal Reports patří:

- Jde o 100% Open Source bez nutnosti zakoupit jakoukoliv licenci a bez omezení možností,
- Dobře zpracovaná serverová komponenta,
- Zvládá jakékoliv množství dat,
- Postaven na C# místo Javy.

Nevýhodami Seal Reports jsou:

- Problémy s podporou konektivity k některým typům databází,
- Absence grafického rozhraní pro návrh reportů,
- Kódování vzhledu jen pomocí CSS nebo NVD3 souborů.

3.4.6. SpagoBI

„SpagoBI suite is an free/open source enterprise-grade software, particularly flexible and adaptable to the end-users' needs, involving a vibrant community. According to its vision, open source software fosters knowledge and expertise sharing, which is a crucial element to offer a really free open source environment." (SpagoBI Labs, 2016)

Firma SpagoBI Labs vyvíjí tento Open Source software ve svých kancelářích v Itálii. Software jako takový je pojatý jako úplný Open Source a firma nenabízí žádné komunitní či placené verze. Vždy tedy dostanete jedinou verzi software, která je volně stažitelná bez jakýchkoliv omezení. Firma postavila politiku SpagoBI na dostupnosti software. To co si pak uživatel může nebo nemusí zaplatit, jsou e-knihy a tutoriály, jak SpagoBI používat. Dále pak nabízí SpagoBI placenou zákaznickou podporu ve třech balíčcích: Bronze, Silver, Gold. Každý balíček obsahuje určitou sortu služeb a firma si ho může zakoupit, pokud plánuje používat SpagoBI na profesionální úrovni.

Co se samotného nástroje týče, jedná se o klasický koncept serverové instance a reportingového nástroje, které jsou doplněny o integrační vrstvu, nástroj o správu metadat a nástroj pro práci s big data. Reportingový nástroj je opět postaven na Eclipse. SpagoBI umí pracovat i s reporty z Birt nebo JasperReports.

SpagoBI tedy obsahuje tyto nástroje:

- SpagoBI Server – server s analytickými nástroji, který umožňuje tvorbu některých jednodušších grafů a reportů přímo v serverové aplikaci.
- SpagoBI Studio – vývojové studio pro tvorbu reportů, dashboardů a grafů, které lze následně odeslat přímo na server, kde jsou neustále přístupné.
- SpagoBI Meta – prostředí pro správu metadat.
- SpagoBI SDK – integrační vrstva, která umožňuje propojení Spago BI s externími nástroji.
- SpagoBI Applications – kolekce vertikálních analytických modelů vytvořených pomocí SpagoBI.

Zhodnocení SpagoBI

SpagoBI je velmi zajímavý nástroj, jehož nespornou výhodou je to, že se opět jedná o 100% open source řešení. Uživatel tedy může používat celé SpagoBI se všemi doplňky absolutně bezplatně. Serverová aplikace poněkud dlouho startuje, nicméně když už se spustí, je

velmi dobře zpracovaná. Uživatel může prezentovat reporty vytvořené ve studiu nebo lze dokonce tvořit jednoduché typy reportů a grafů přímo v serverovém rozhraní pomocí modulu zvaného Cockpit. Jedná se o jeden z nejvhodnějších nástrojů pro řešení našeho problému.

Výhody SpagoBI jsou:

- Jedná se o 100% Open Source bez nutnosti zakoupit jakoukoliv licenci a bez omezení možností,
- Dobře zpracovaná serverová komponenta, která umožňuje nejen prezentaci, ale také interaktivní tvorbu reportů přímo v rozhraní serveru,
- Zvládá jakékoliv množství dat,
- Přizpůsobení pro mobilní zařízení. Server detekuje připojení z mobilního zařízení a automaticky mu přizpůsobuje vzhled rozhraní,
- Umožňuje jednoduché propojení s ETL nástrojem Talend OS

Nevýhody SpagoBI jsou:

- Téměř žádná zákaznická podpora, pokud si zákazník nezaplatí,
- Problémy s podporou konektivity k JDBC databázi,
- Komplikovaná instalace a spouštění.

3.5. Zúžení výběru vhodných OS nástrojů

Všechny BI nástroje zmíněné v minulé kapitole byly několik týdnů testovány. Sledovali jsme především soulad možností nástrojů s požadavky v zadání projektu. Dále pak hrálo roli, jaké možnosti software nabízí bez nutnosti platit licenci. Neméně důležitým hlediskem bylo, jak výsledné reporty a datové vizualizace vypadají a jak široké jsou možnosti nastavení jejich vzhledu. Významným faktorem byla také uživatelská přívětivost, složitost instalace a zprovoznění celé aplikace.

Výsledkem je redukce počtu původně navrhovaných OS nástrojů z šesti na tři.

Tyto tři nástroje jsou:

- Birt,
- Pentaho,
- SpagoBI.

Výše zmíněné tři nástroje se tedy jevily jako nejvhodnější kandidáti pro realizaci projektu. Především z důvodu kvalitního zpracování celého software a velkého množství možností využití v open source verzi.

JasperReports byl vyřazen hlavně kvůli neúplné funkčnosti některých stěžejních funkcionalit v OS verzi.

Co se týče Jedox Base, nástroj byl vyřazen především kvůli jeho závislosti na MS Excel, který má placenou licenci a tudíž nutí uživatele Excel zakoupit. Avšak nebyl to zdaleka jediný problém. Zaznamenali jsme velké množství problémů některých základních funkcionalit a problémů při navázání databázové konektivity, způsobených tímto software.

Seal Reports byl vyřazen zejména kvůli absenci grafického rozhraní pro tvorbu vizualizací, která komplikovala celý vývoj reportů a byla velmi uživatelsky nepřívětivým faktorem. Celkově se tento nástroj jeví jako velmi nedodělaný a některé z klíčových funkcionalit nejsou dotaženy dokonce.

Tři nástroje, které uspěly v užším výběru, byly podrobeny finální selekci. Z této selekce vzejde jeden „vítězný“ nástroj, ve kterém bude následně projekt zrealizován.

4. Testování zúženého výběru dle firemních požadavků a jeho vyhodnocení

Tato část bude věnována výběru finálního nástroje, pomocí kterého budeme tvořit řešení celého projektu. Z tří „finalistů“ jmenovitě Birt, Pentaho a SpagoBI vybereme takového, jež bude pro projekt nejvhodnější.

Celý výběr bude probíhat pomocí dotazníku vytvořeného pro získání feedbacku třetích stran. V tomto případě se bude jednat o třicet BI specialistů z business intelligence oddělení firmy Tieto. Jde především o odborníky v oblasti ETL a reportingu a mají tudíž obsáhlé znalosti této problematiky.

Ještě jednou zdůrazníme požadavky zákazníka, které se týkají především toho, aby celé řešení bylo minimálně nákladné. Dále pak aby bylo řešení online přístupné na jednom ze serverů Tieto a bylo mnohem automatizovanější než doposud. Řešené vizualizace by také měly vyhovovat zákazníkovu firemnímu barevnému schématu.

4.1. Sestavení dotazníku

Bude se jednat o krátký dotazník, který má zachycovat otázky zaměřené na přednosti zvolených softwarových nástrojů. Otázky jsou tedy sestaveny jednoduše, věcně a přehledně. Jedná se o otázky postavené zejména na poznatcích získaných v předchozí kapitole, které se týkají výběru nástrojů pro OS business intelligence řešení.

Ze všech získaných poznatků jsme vytvořili devět otázek, na něž odpovídalo 30 BI specialistů, kteří byli s každým ze tří výsledných software seznámeni v rámci několikahodinového předváděcího meetingu. Na tomto meetingu byly demonstrovány možnosti každého z navrhovaných SW kandidátů. BI specialisté pracují s podobnými softwarovými nástroji běžně a mají s nimi velké zkušenosti. Proto může být jejich erudovaný názor pro další rozhodnutí relevantní. Každá z otázek se dotýká určité oblasti BI řešení a měla by sloužit ke zhodnocení toho, jak kvalitně daný software tuto oblast pokrývá.

Finální znění otázek vypadá takto:

1. Který software má největší rozsah možností v OS verzi?
2. Který software má nejlépe řešený nástroj pro tvorbu reportů?
3. Který software má nejlépe řešený ETL nástroj?
4. Který software umožňuje nejefektivnější správu metadat?
5. Který software má nejlépe řešené serverové rozhraní?
6. Který software je schopen integrovat data z nejvíce různorodých zdrojů?
7. Který software je uživatelsky nejpřívětivější?
8. Ve kterém software vám připadá vzhled vizualizací nejatraktivnější?
9. Který software má nejlepší uživatelskou podporu?

U každé z otázek byly pouze tři možnosti odpovědi:

a) Birt,
b) Pentaho,
c) SpagoBI.

4.2. Váhy otázek

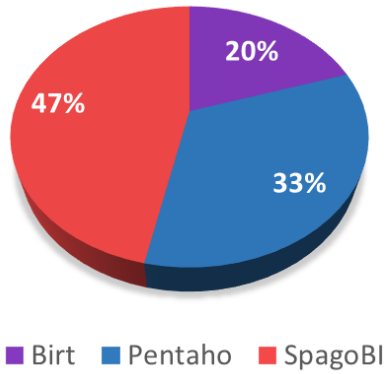
Pro lepší vypovídací hodnotu dotazníku jsme zavedli váhové ohodnocení otázek podle jejich vlivu na splnění požadavků zadavatele. Váhy nabývají hodnot 1 – 5, kde 5 má největší

vliv a 1 nejmenší. Jelikož je naše řešení zaměřeno především na reporting, vizualizace a OS reportingové nástroje, budou mít otázky týkající se přímo této oblasti největší váhu.

4.3. Vyhodnocení dotazníku

Tato podkapitola bude věnována vyhodnocení našeho BI dotazníku. Výsledky jednotlivých otázek budeme vyobrazovat pomocí koláčových grafů. Nakonec provedeme souhrn všech výsledků a vybereme finální nástroj pro realizaci řešení.

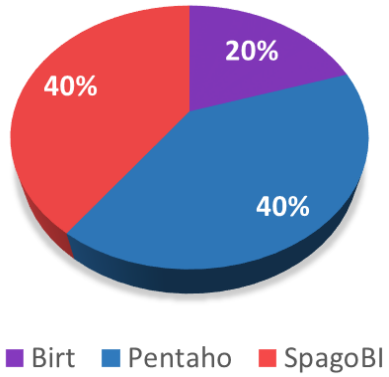
Který software má největší rozsah možností v OS verzi?



Obrázek 4.1. Graf výsledků otázky č.1., zdroj: vlastní zpracování

Z grafu můžeme vidět, že ohledně možností v OS verzi jednoznačně uspěl software SpagoBI s počtem čtrnácti respondentů, což tvořilo 47% z celku. Podle BI specialistů má tedy tento nástroj nejširší rozsah možností v OS licenci. Těsně za ním skončilo Pentaho a na posledním místě Birt.

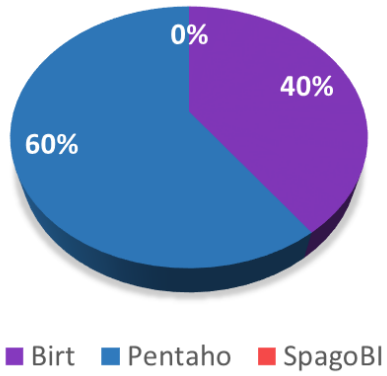
Který software má nejlépe řešený nástroj pro tvorbu reportů?



Obrázek 4.2. Graf výsledků otázky č.2., zdroj: vlastní zpracování

Co se nástroje pro tvorbu reportů týče, Pentaho a Spago BI získaly shodný počet dvanácti hlasů, což je 40% z celku. Jejich reportingové nástroje jsou tedy podle odborníků srovnatelné, zatímco Birt dostal jen 6 hlasů, což je 20% z celku.

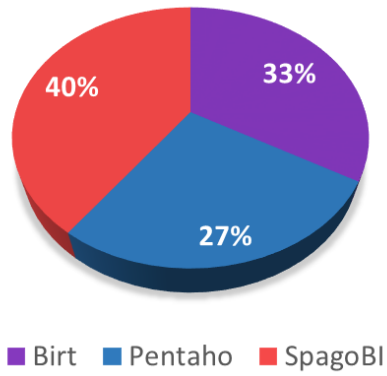
Který software má nejlépe řešený ETL nástroj?



Obrázek 4.3. Graf výsledků otázky č.3., zdroj: vlastní zpracování

V kvalitě ETL nástroje jednoznačně vítězí Pentaho se svým nástrojem Pentaho Data Integration, který dostal 60% z celkových třiceti hlasů. Nejhůře dopadlo SpagoBI, které dostalo 0% hlasů a to proto, že SpagoBI jako takové žádný vlastní ETL nástroj nemá. Může však pro ETL využít OS nástroj Talend Open Studio.

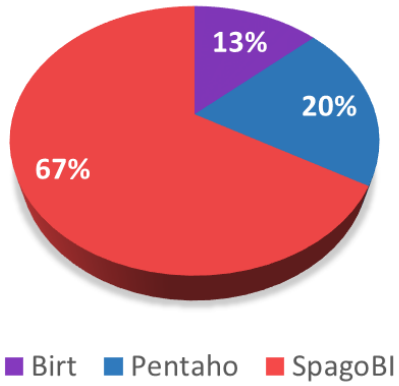
Který software umožňuje nejefektivnější správu metadat?



Obrázek 4.4. Graf výsledků otázky č.4., zdroj: vlastní zpracování

V oblasti nástrojů pro správu metadat je Pentaho na posledním místě s 27% hlasů. Vítězí nástroj SpagoBI Meta, který dostal 40% hlasů, jelikož nástroj SpagoBI Meta byl BI specialisty hodnocen velmi pozitivně a jeho možnosti jsou na OS BI nástroj velmi rozsáhlé.

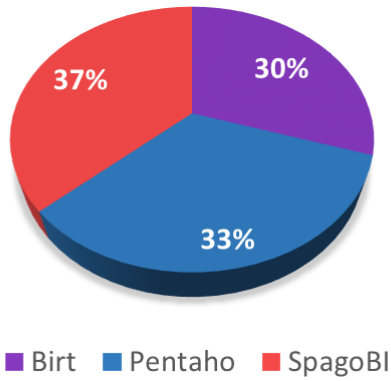
Který software má nejlépe řešené serverové rozhraní?



Obrázek 4.5. Graf výsledků otázky č.5., zdroj: vlastní zpracování

V otázce zaměřené na kvalitu serverové instance a jejího rozhraní na celé čáře zvítězil nástroj SpagoBI. Získal 67% ze všech hlasů. Zaujala zejména možnost tvorby reportů a grafů přímo v serverovém rozhraní pomocí jednoduché drag and drop metody a také celkové zpracování včetně politiky uživatelských práv atd.

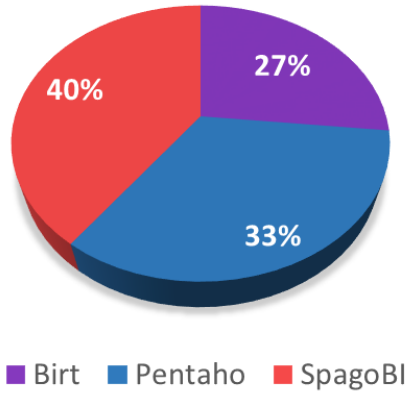
Který software je schopen integrovat data z nejvíce různorodých zdrojů?



Obrázek 4.6. Graf výsledků otázky č.6., zdroj: vlastní zpracování

V této otázce byly všechny tři software velmi vyrovnané. O jeden hlas zvítězilo SpagoBI, druhé skončilo Pentaho a o jeden hlas jako třetí skončil Birt. Je pravdou, že možnosti integrace různých typů datových zdrojů jsou v každém z testovaných software velmi obsáhlé, což je nejspíš i důvodem tohoto výsledku.

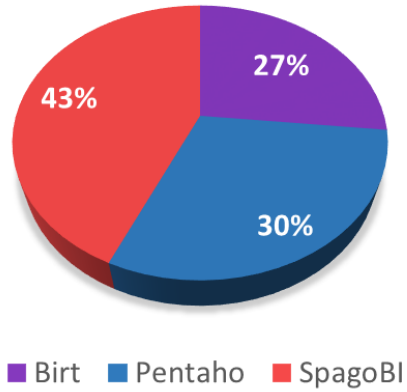
Který software je uživatelsky nejpřívětivější?



Obrázek 4.7. Graf výsledků otázky č.7., zdroj: vlastní zpracování

Co se uživatelské přívětivosti týče, zvítězil software SpagoBI. Tento software je tedy podle BI specialistů nejvíce user friendly, což je také důležitý aspekt, ke kterému je třeba přihlížet.

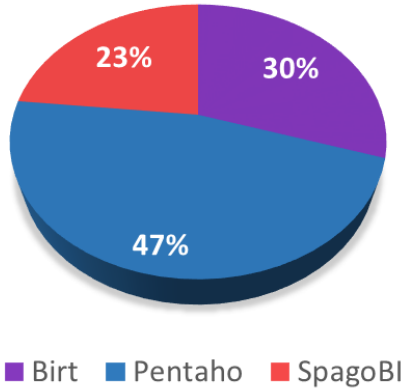
Ve kterém software vám připadá vzhled vizualizací nejatraktivnější?



Obrázek 4.8. Graf výsledků otázky č.8., zdroj: vlastní zpracování

V atraktivnosti a pěkném vzhledu výsledných reportů zvítězilo opět SpagoBI se 43% hlasů. Většina z hlasujících to odůvodňovala integrací Java Highcharts, což umožňuje vykreslování velmi dobře vypadajících responzivních grafů, jež reagují a mění svůj layout podle displeje zařízení, ze kterého jsou vyobrazovány.

Který software má nejlepší uživatelskou podporu?



Obrázek 4.9. Graf výsledků otázky č.9., zdroj: vlastní zpracování

Z obrázku 4.9. je patrné, že co se týče kvality uživatelské podpory, poměrně jednoznačně vítězí Pentaho, které má kvalití komunitní web se spoustou tutoriálů a radami jak Pentaho využívat. Proto získalo 47% hlasů. Birt skončil na druhém místě s 30% hlasy. Jeho podpora není až tak rozsáhlá jako u Pentaha, ale přesto se dá najít velké množství Birt tutoriálů a ukázek. Nejslabší je z tohoto ohledu SpagoBI, kde je podpora bez zaplacení jednoho z firmou nabízených balíčků téměř nulová. Existuje sice Spago World Forum, ale to je velmi málo navštěvováno.

4.4. Finální výběr nástroje pro realizaci řešení

Ze tří námi vybraných softwarových nástrojů je tedy třeba vybrat finální nástroj, ve kterém bude projekt realizován. V tabulce 4.1. můžeme vidět přehled počtu odpovědí pro všech devět otázek. Číselné hodnoty v polích znázorňují kolik BI specialistů hlasovalo pro danou možnost.

Otázka	Birt	Pentaho	SpagoBI	Váha
1.	6	10	14	5
2.	6	12	12	5
3.	12	18	0	2
4.	10	8	12	3
5.	4	6	20	4
6.	9	10	11	3
7.	8	10	12	4
8.	8	9	13	5
9.	9	14	7	1

Tabulka 4.1. Přehled výsledků dotazníku, zdroj: vlastní zpracování

Modrá pole značí, že daný software získal za tuto otázku největší počet hlasů. Fialový sloupec vyjadřuje hodnoty vah jednotlivých otázek. Na první pohled tedy můžeme vidět, že Birt nedostal největší počet hlasů ani v jedné z devíti otázek. Je tedy podle dotazníku nejslabším ze všech testovaných software.

Pentaho dostalo největší počet hlasů ve dvou otázkách. V otázkách tři a devět, které se týkaly nejkvalitnějšího ETL nástroje a rozsahu podpory daného software. Takže v těchto dvou oblastech Pentaho podle názoru BI specialistů dominuje nad ostatními software, ovšem obě tyto otázky měly nízkou váhu. V otázce číslo dvě s váhou 5, jež se týkala kvality nástroje pro tvorbu reportů, dostalo Pentaho stejné množství hlasů jako SpagoBI. Dalo by se tedy říct, že co se kvality týče, jsou oba reportingové nástroje srovnatelné.

SpagoBI se podle dotazníku jeví jako absolutní vítěz. Tento software dostal největší počet hlasů v šesti z devíti otázek. Jednalo se převážně o otázky s velkou váhou, konkrétně dvě otázky s váhou 5, dvě otázky s váhou 4 a taktéž dvě otázky s váhou 3. Podle názoru BI specialistů z firmy Tieto nabízí tento software v OS verzi nejvíce ze všech testovaných software. Má nástroj pro tvorbu reportů srovnatelné kvality jako Pentaho. Má také velmi kvalitní nástroj na správu metadat. Serverové rozhraní SpagoBI porazilo všechny ostatní na celé čáře. Je schopen integrovat data z mnoha různorodých zdrojů a je velmi uživatelsky přívětivý. Jedním z nejdůležitějších faktorů je také, že poskytuje velmi dobře vypadající datové vizualizace.

Jedinou oblastí, ve které SpagoBI selhalo, je ETL nástroj. Jelikož SpagoBI samo žádný ETL nástroj neobsahuje. Nicméně podporuje propojení se známým ETL nástrojem Talend Open Studio, který je při OS řešení ETL hojně využíván. Otázka týkající se ETL nástrojů měla nízkou váhu 2 právě kvůli dostupnosti relativně kvalitních OS ETL nástrojů jako Talend a není tudíž stěžejní, aby měl reportingový nástroj vyloženě vlastní ETL řešení.

Pro realizaci projektu bylo tedy vybráno SpagoBI, s tím, že pro ETL úkony využijeme služeb Talend Open Studio. Tyto nástroje, společně s MS SQL Server, by měly postačovat k realizaci celého našeho projektu.

5. Implementace nejvhodnějšího řešení

Tato kapitola bude věnována postupné implementaci našeho BI řešení pomocí SQL Server Management Studio, které bude sloužit pro vytvoření cílové datové struktury a datového modelu. Dále využití Open Source nástroje Talend Open Studio pro samotnou extrakci,

transformaci a nahrání dat získaných z Jiry. V poslední části se budeme věnovat využití OS nástroje SpagoBI, jež byl vybrán v minulých kapitolách, pro tvorbu automatizovaného dynamického reportingu.

5.1. Tvorba datového modelu

Datový model byl v prvé řadě prokonzultován se zadavatelem a s pracovníky, kteří doposud na tomto projektu pracovali. Jak už bylo řečeno na začátku práce, přímý přístup ke zdrojové databázi bohužel kvůli problematickému zřizování nebylo možno získat. Proto jsou data jednou měsíčně exportovány pomocí javascriptové aplikace přímo z Jiry do XLSX souboru.

Takovéto XLSX soubory jsou celkem tři. První část dat se týká údajů o chybách na serverech. Další se týká serverových incidentů a poslední se týká user story (uživatelských příběhů).

V souvislosti s tím je tedy nutno vytvořit databázi s třemi nezávislými tabulkami, jež budou reflektovat data sloužící k reportingu jednotlivých oblastí. Tyto tabulky vytvoříme ve známém vývojovém prostředí od Microsoftu MS SQL Server Management studio. První tabulka je tedy tblBugReports, která bude určena pro data o chybách. Dále tabulka tblIncidents, jež bude sloužit pro ukládání dat o incidentech a poslední tabulkou bude tblExportStory, která slouží k uchování dat o user story.

tblBugReports		
Název sloupce	**Datový typ**	**Null**
ID_Bug	varchar(100)	Ne
IssueType	varchar(3)	Ne
Priority	varchar(10)	Ne
Status	varchar(30)	Ne
Resolution	varchar(30)	Ne
DevTeam	varchar(20)	Ano
CreateTime	date	Ne
FixVersions	varchar(50)	Ano
AffEnv	varchar(10)	Ano
AffVersions	varchar(50)	Ano
Components	varchar(100)	Ano

Tabulka 5.1. tblBugReports, zdroj: vlastní zpracování

Z tabulky 5.1 můžeme vidět přehled atributů pro tblBugReports. Primární klíče budou v tabulce vyznačeny světle fialovou barvou. Tato tabulka se týká především bugů. Obsahuje

atributy jako Priority, což je priorita chyby, dále například AffEnv, což je informace o prostředí, ve kterém se chyba vyskytuje a v neposlední řadě také ID_Bug, což je konkrétní číslo chyby a tudíž i primární klíč tabulky.

tblExportStory		
Název sloupce	**Datový typ**	**Null**
ID_Story	varchar(100)	Ne
Status	varchar(10)	Ne
FixVersions	varchar(50)	Ano
CreateTime	date	Ne
ClosedTime	date	Ne
UserStoryType	varchar(14)	Ne
OriginalEstimate	float	Ano
TimeSpent	float	Ano
E2EDQ	int	Ne
E2ETime	int	Ne
FinalDQ	int	Ano

Tabulka 5.2. tblExportStory, zdroj: vlastní zpracování

Tabulka 5.2 tblExportStory tedy obsahuje atributy, které se týkají uživatelských scénářů. Atribut ID_Story je číslo konkrétního incidentu, je to zároveň primární klíč a proto je tedy vyznačen fialovou barvou. Dále obsahuje např. atribut UserStoryType, což je typ scénáře. Poté zde můžeme najít například atributy jako CreateTime, ClosedTime, které obsahují údaj o času vytvoření scénáře.

tblIncidents		
Název sloupce	**Datový typ**	**Null**
ID_Inc	varchar(8)	Ne
OpenTime	datetime	Ne
Status	varchar(10)	Ne
ProbStatus	varchar(25)	Ne
SeverityCode	int	Ne
Severity	varchar(8)	Ne
PriorityCode	int	Ano
Priority	varchar(8)	Ano
ResolvedTime	datetime	Ano
Assignment	nchar(100)	Ano

Tabulka 5.3. tblIncidents, zdroj: vlastní zpracování

Poslední tabulkou je tblIncidents, která se týká dat o incidentech. Primárním klíčem je zde ID_Inc, což je konkrétní a unikátní číslo incidentu. Dále je zde několik atributů jako například Severity, což je závažnost incidentu nebo ResolvedTime, který označuje čas, kdy byl incident vyřešen.

Pomocí MS SQL Server Management studio byla tedy na serveru Tieto vytvořena databáze, ve které byly vytvořeny všechny tři výše zmíněné tabulky. Do této databáze budou periodicky nahrávána data získaná z Jiry.

5.2. Realizace ETL pomocí Talend Open Studio

Po vytvoření databáze a datového modelu je nyní třeba databázi naplnit. K tomu poslouží Open Source software Talend Open Studio, který lze přímo propojit i s reportingovým nástrojem SpagoBI. Využijeme tedy možností tohoto poměrně rozsáhlého Open Source ETL nástroje.

5.2.1. Talend Open Studio

Jak už bylo řečeno výše, ETL bude realizováno pomocí software TOS. V této části specifikujeme některé z možností, jež tento nástroj nabízí.

„Talend's open source products and open architecture create unmatched flexibility so you can solve integration challenges your way. Talend reduces the learning curve and lowers the barrier to adoption for data integration, data profiling, big data, application integration, and more." (Talend, 2016)

Jedná se o specializovaný nástroj, který je zaměřen na ETL a datovou integraci. Pomocí Talendu je možno vytvořit business modely, nebo ETL úkony a je také možno spravovat jejich plánové spuštění. Dále můžeme integrovat data z velkého množství různorodých datových zdrojů. Buď přímo z databází, nebo z XLSX, XML a mnoha dalších typů souborů. Poté je lze přes komponenty typu map namapovat do struktury cílové databáze a následně provést nahrání.

5.2.2. Tvorba úloh v TOS

Celý proces by se dal rozdělit do tří úloh pro nahrání do tří cílových tabulek. Jelikož datový rozsah není příliš velký a tabulky jsou periodicky aktualizovány jednou měsíčně všechny zároveň, bylo možné spojit tyto tři úlohy do jedné společné úlohy.

Vytvořili jsme tedy úlohu s názvem Bug_Report, která se bude starat o extrakci původních dat z XLS souborů, jejich následnému přizpůsobení a nahrání do tabulek cílové

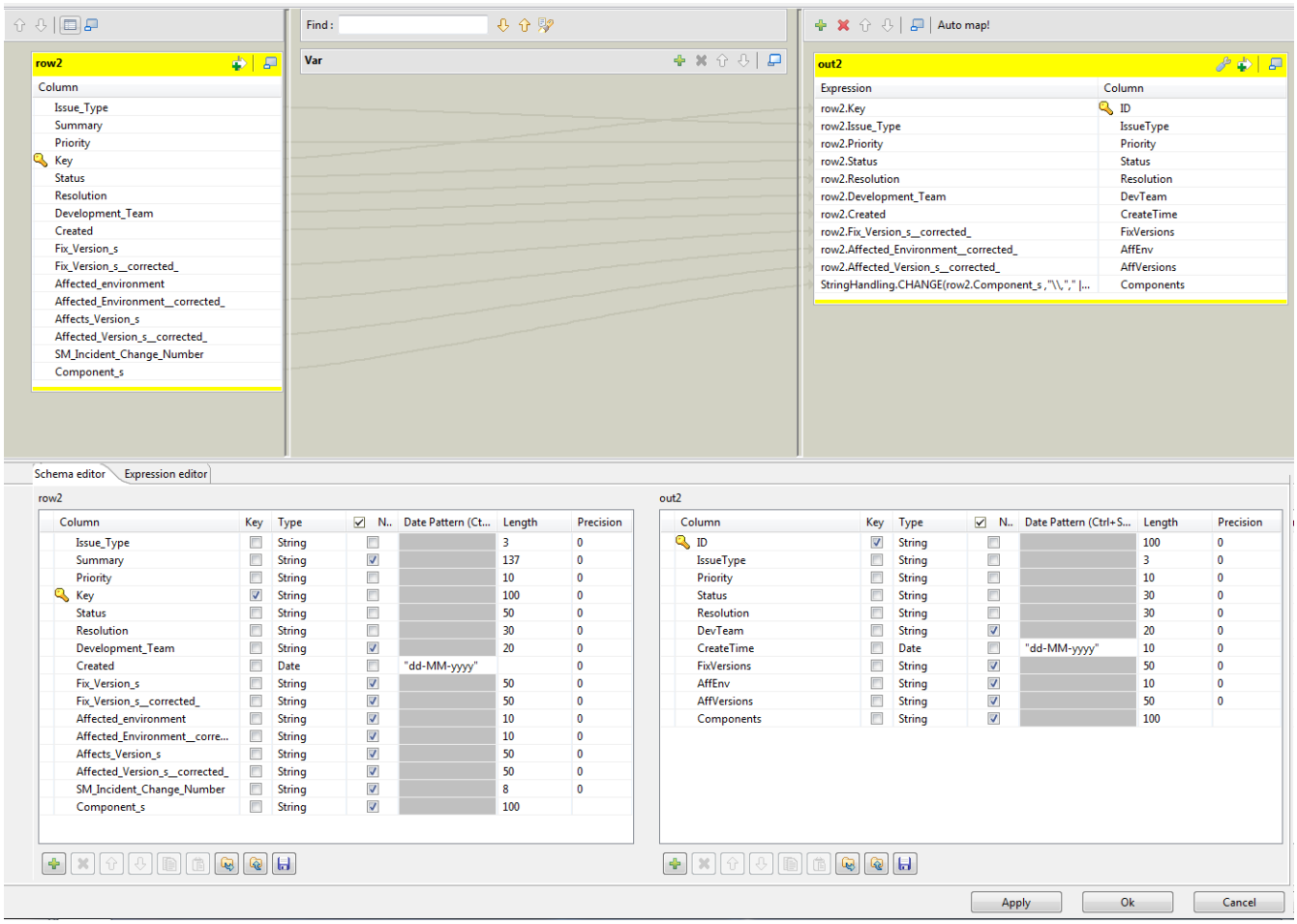
databáze. Než začneme pracovat na samotném zpracování úlohy, je nutné v sekci Db Connections nastavit připojení k naší cílové databázi.

Vstupní komponenty Excel file

Nejprve je třeba vytvořit tři komponenty typu tFileInputExcel, které slouží k získání dat z XLS a XLSX souborů. Každá tato komponenta bude pro jeden z našich zdrojových XLXS souborů. Vytvořili jsme tedy komponenty ExportStory, BugReports a Incidents, ve kterých byla nastavena cesta na zdrojový adresář obsahující Excel soubory, periodicky získávané exportem z Jiry. Z tohoto adresáře budou získávána zdrojová data pro jednotlivé komponenty při každém spuštění úlohy.

Mapovací komponenty

Pro namapování zdrojových tabulek do tabulek v databázi byly vytvořeny tři komponenty typu tMap, které slouží k přípravě dat pro cílovou databázi s případnými transformacemi nebo redukcemi nepotřebných sloupců.



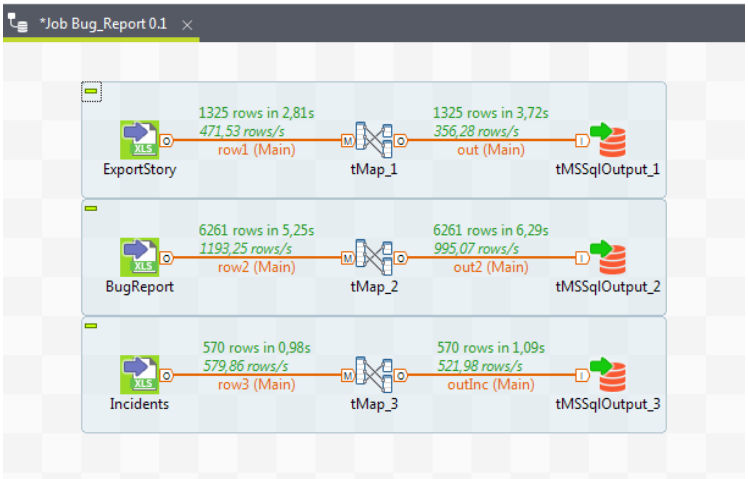
Obrázek 5.1. Mapování tabulky BugReports, zdroj: vlastní zpracování

Na obrázku můžeme vidět grafické rozhraní pro samotný proces mapování tabulky BugReports. Po otevření komponenty tMap se zobrazí okno, ve kterém mapujeme sloupce ze vstupní tabulky do cílové tabulky. Jak je vidět z obrázku nejsou využity všechny sloupce, ale jen ty, které jsou pro naše reporty skutečně potřebné. Dochází také k drobným úpravám dat. Například v sloupci Components dochází pomocí funkce StringHandling CHANGE ke změně původních hodnot, kde je čárka nahrazena za znak pipe | neboli hovorově „roura“. Důvod je čistě formální, jelikož reportingový software SpagoBI vnímá čárku jako znak pro oddělení parametrů. Pokud by tedy čárka nebyla nahrazena, vnímal by jeden dlouhý název komponenty s čárkou uprostřed jako dvě odlišné komponenty a případné filtrace nebo agregační funkce komponent by tedy měly odlišné výsledky.

Dále zde lze nastavit datové typy jednotlivých sloupců, stejně jako přesný formát pro sloupce typu Date. Dle potřeby zde také nastavíme, zda mohou být hodnoty atributů nulové či nikoliv a také Lenght, což je počet znaků, který může stringový parametr maximálně mít. Po úspěšném namapování stačí potvrdit a zavřít okno. Podobným způsobem probíhá i mapování ostatních tabulek.

Výstupní komponenty MS SQL Server

Všechny tři mapovací komponenty ústí přímo do výstupních komponent typu tMSSqlOutput. Jak už název napovídá, tyto komponenty slouží pro napumpování dat do MS SQL Serveru. Je třeba u nich nastavit specifikaci výstupního serveru a příslušné přístupové údaje. Mapovací komponenty získají z výstupních komponent údaje o struktuře a schématu výstupní databáze. U všech tří komponent je na výstupu nastaven příkaz update, který při spuštění tabulky aktualizuje.



Obrázek 5.2. Grafické vyobrazení komponent v TOS při spuštění úlohy, zdroj: vlastní zpracování

Po spuštění úlohy tedy proběhne celý proces a data se začnou nahrávat do cílových tabulek, což indikuje informace o tom, kolik řádků za sekundu bylo nahráno do databáze. Konzole poté hlásí, že celé nahrávání proběhlo v pořádku, případně vypisuje červenou barvou chyby, kvůli kterým nebylo možno nahrávání provést.

Tím je vyřešena celá ETL část a naše data jsou připravena v cílové databázi na serveru Tieto. Z takto upravených dat můžeme začít tvořit dynamické reporty pomocí reportingového nástroje.

5.3. Řešení reportingu a vizualizace dat pomocí SpagoBI

Tato finální část obsahuje konkrétní popis řešení automatizovaného dynamického reportingu našich dat pomocí SpagoBI. Všechny hodnocené software byly Open Source verze s licencí zdarma a jak už bylo řečeno výše, SpagoBI není výjimkou, což koresponduje s požadavkem minimálních nákladů na celý projekt. Zákazník také dodal předlohy většiny požadovaných reportů a dashboardů, které by chtěl realizovat ve formě excel souboru, kde jsou reporty sestrojeny pomocí MS Power Pivot.

Nejdříve byly zprovozněny všechny součástí SpagoBI. Výhodou je, že instalace jako taková není nutná. SpagoBI dodává nástroje ve formě ZIP souboru, který stačí extrahovat a nástroje již fungují bez instalace. Bylo však potřebné stáhnout a nainstalovat poslední verzi JDK (Java Development Kit) z oficiálního webu firmy Oracle. Poté byla v operačním systému vytvořena proměnná uživatelského účtu s názvem JAVA_HOME odkazující na konkrétní adresář s instalací JDK. Pak bylo možno SpagoBI bez problému spustit.

5.3.1. Selekce dat

Po instalaci, než jsme začali ve SpagoBI Studio tvořit samotné reporty a vizualizace, bylo nutno nejdříve vytvořit datasety (datové sady) na SpagoBI serveru, které pak budou naše vizualizace vyobrazovat.

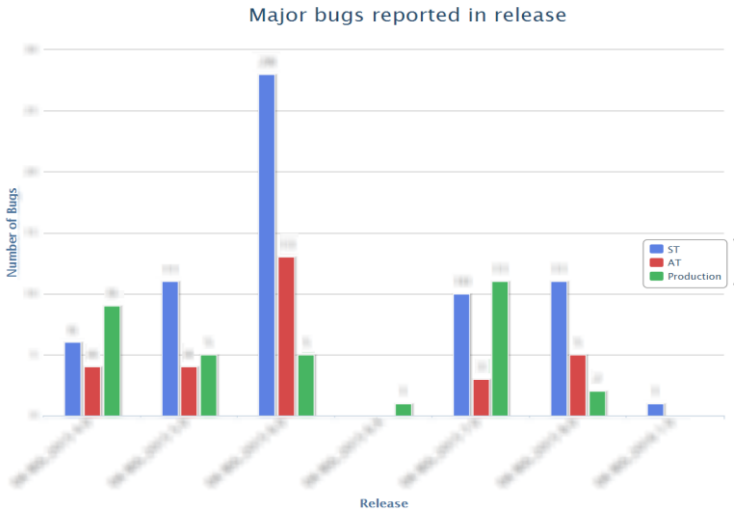
Museli jsme tedy nejdříve spustit SpagoBI server a v serverovém rozhraní navázat konektivitu s naší databází na Tieto serveru. Klasicky pomocí připojovacího stringu (connection string) a přihlašovacích údajů (credentials).

Poté jsme mohli přejít k samotné tvorbě datasetů pro jednotlivé reporty. Datasety se tvoří pomocí selectů napsaných v jazyce SQL na SpagoBI serveru. SpagoBI neumí dělat klasické agregace dat v návrhu reportů. Agregace v rámci reportů lze ve SpagoBI dělat jen pokud se

jedná o reporty typu Birt. Je tedy třeba mít data správně agregovaná již v rámci datasetu. Musíme tak využívat podmínek a klauzulí where.

```
Select AffVersions,
NULLIF(COUNT(CASE WHEN AffEnv = 'ST' THEN ID_Bug
ELSE null
END), 0) AS ST,

NULLIF(COUNT(CASE WHEN AffEnv = 'AT' THEN ID_Bug
ELSE null
END), 0) AS AT,

NULLIF(COUNT(CASE WHEN AffEnv = 'Production' THEN ID_Bug
ELSE null
END), 0) AS Production,

NULLIF(COUNT(CASE WHEN AffEnv = 'PT' THEN ID_Bug
ELSE null
END), 0) AS PT


from tblBugReports
where
Priority ='Major'


Group By AffVersions

Order By AffVersions
```

V kódu výše můžeme vidět jeden z ukázkových selectů pro report, jež má reprezentovat počet bugů, které vznikly v jednotlivých typech environmentů (prostředí). Jedná se o prostředí ST, AT, Production a PT. Pomocí COUNT jsou spočítány bugy v jednotlivých prostředích. COUNT je ještě zároveň obalen v příkazu NULLIF, který je proveden jen v případech, že data nabývají hodnoty 0. NULLIF je zde použit proto, že ve výsledném grafu, pro který je tento select zdrojem, nechceme vykreslovat hodnoty s nulou. Proto se v případech, kdy nabývají hodnoty 0, nahrazují za null.



Obrázek 5.3. Major bugs reported in release, zdroj: vlastní zpracování

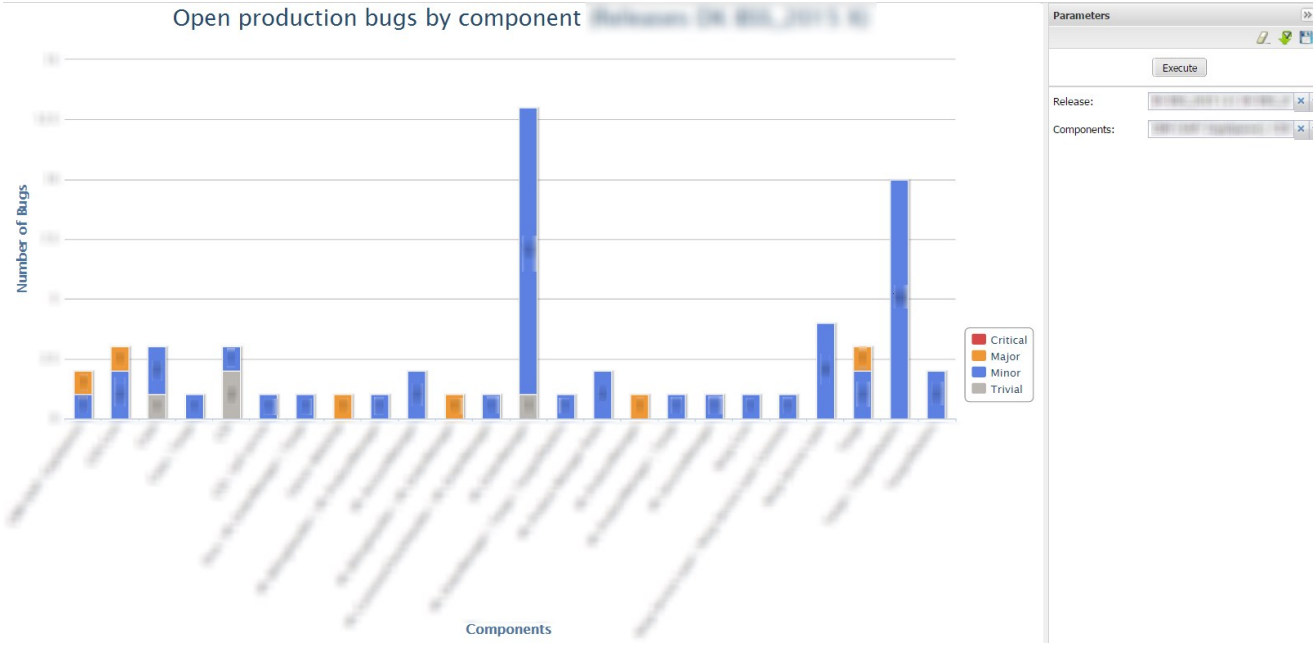
Na obrázku 5.3. můžeme vidět náhled výsledného grafu, jehož plné vyobrazení je v příloze č. 2. Jak už bylo uvedeno v úvodu kapitoly 5., všechna data jsou z důvodu bezpečnosti znečitelněna na přání zákazníka (v tomto i následujících grafech). Tento graf má sloužit k vyobrazování bugů s Major prioritou. Takže je v klauzuli where ještě podmínka, že priorita musí být Major. Celý dataset je ještě seskupen a seřazen na ose X podle AffVersions, což jsou jednotlivé verze releasů, kterých se bugy týkají. S každou novou verzí releasu je vždy znova měřen počet bugů v tom daném releasu. Podobným způsobem jsou tvořeny všechny datasety pro požadované reporty a dashboardy.

5.3.2. Parametrizace datasetů

Abychom neměli jen statické grafy, je třeba jednotlivé datasety parametrizovat. Za parametry potom pomocí roletky, checkboxu nebo vyskakovacího okna můžeme dosadit námi zvolené hodnoty, report se s nimi obnoví a bude filtrován podle uživatelem zvolených možností. Nejprve je však nutno parametrizovat samotný kód selectu pro dataset, což si ukážeme v následujícím příkladovém selectu.

```
Select Components,

NULLIF(COUNT(CASE WHEN Priority = 'Critical' THEN ID_Bug
ELSE null
END), 0) AS Critical,

NULLIF(COUNT(CASE WHEN Priority = 'Major' THEN ID_Bug
ELSE null
END), 0) AS Major,

NULLIF(COUNT(CASE WHEN Priority = 'Minor' THEN ID_Bug
ELSE null
END), 0) AS Minor,

NULLIF(COUNT(CASE WHEN Priority = 'Trivial' THEN ID_Bug
ELSE null
END), 0) AS Trivial,

NULLIF(COUNT(CASE WHEN Priority = 'Blocker' THEN ID_Bug
ELSE null
END), 0) AS Blocker

from tblBugReports
where

AffEnv ='Production'
AND
Status in ('AT Tested','In Analysis','In Development','Merged',
'Open','Ready for AT','Ready for Merge','Ready for ST','ST tested','ST Testing')
AND
AffVersions In ($P{Release})
AND
Components In ($P{Comp})

Group By Components

Order By Components
```

Jedná se o select, který je pro graf vyobrazující serverové komponenty. Je zde opět mnoho SQL příkazů jako NULLIF, COUNT atd., všechny tyto jsme zmínili už v minulé podkapitole. To, co je zde však stěžejní, je parametrizace v klauzuli where. Ta se ve SpagoBI tvoří pomocí syntaxe $P{Jméno_parametru}. Můžeme tedy vidět, že v reportu jsou konkrétně dva parametry $P{Release} a $P{Comp}. Parametr Release slouží k parametrizaci releasů, za které budou bugy v selectu sčítány a parametr Comp slouží k parametrizaci komponent, jež jsou v grafu vyobrazeny na ose X. Jedná se o skládaný sloupcový graf, takže data budou rozdělena podle toho, jakou mají bugy prioritu.



Obrázek 5.3. Open production bugs by componet, zdroj: vlastní zpracování

Na tomto obrázku je výsledný skládaný sloupcový graf, týkající se množství bugů podle komponent, u kterých se vyskytly. V plné velikosti můžeme tento graf nalézt v příloze č. 3. Na pravé straně jsou filtrace, jež jsme vytvořili právě díky parametrizaci selectu pro příslušný dataset. Samotné tvorbě a nastavení filtrací je věnována samostatná podkapitola.

5.3.3. Tvorba vizualizací a reportů

Po vytvoření potřebných datasetů jsme přešli k samotnému návrhu reportů a grafů v SpagoBI Studio. Hned po vytvoření nového projektu je třeba ve složce resources v podsložce server navázat konektivitu s konkrétní instancí SpagoBI serveru, která slouží pro deploy vytvořených reportů. To je pro účely vývoje reportů server spuštěný na naší pracovní stanici.

Následně můžeme začít tvořit jednotlivé reporty. Přímo ve SpagoBI Studio je možno tvořit i business modely a návrhy OLAP kostek, nicméně těchto možností v naší práci nevyužíváme. Soustředíme se na čistý návrh reportů, grafů a dashboardů.

Všechny grafy jsou realizovány pomocí Java Highcharts, jejichž knihovna používá tzv. JSON (JavaScript Object Notation) šablonu. Tato šablona popisuje objekty pomocí XML (Extensible Markup Language), je tedy nutno tvořit grafy pomocí XML kódu. Tento kód je složen z tagů podobně jako HTML. Nicméně se jedná o speciální tagy, které umí číst SpagoBI Server. SpagoBI nabízí i grafické rozhraní pro kódování XML, kde je možno nastavovat v roletkách parametry objektů. Avšak toto rozhraní není úplně dokonalé a nelze v něm nastavit vše. Proto je výhodnější kódovat šablony objektů ručně přímo v textovém XML souboru.

V adresáři SpagoBI projektu byly tedy vytvořeny tři složky pro každou sadu objektů z jedné tabulky. Jsou to složky BugReports, ExportStory a Incidents. V každé z těchto složek byl vytvořen nespočet objektů. Kódování některých z nich si zde pro příklad uvedeme.

Major bugs reported in release

```
<HIGHCHART width="100%" height="100%">
  <CHART defaultSeriesType="column" />
  <TITLE text="Major bugs reported in release" fontSize="25" >
   </TITLE>
  <LEGEND layout="vertical" align="right" verticalAlign="middle" />
  <X_AXIS categories="AffVersions" alias="AffVersions" fontSize="30" >
       <LABELS rotation="315">
           <STYLE fontSize="15" />
        </LABELS>
     <TITLE text="Release">
     <STYLE fontSize="15" />
     </TITLE>
  </X_AXIS>

    <Y_AXIS>
       <TITLE text="Number of Bugs">
       <STYLE fontSize="15" />
     </TITLE>
    </Y_AXIS>

  <PLOT_OPTIONS>
    <COLUMN animation="true" enableMouseTracking="true" shadow="true"
showInLegend="true" visible="true">
      <DATA_LABELS  enabled="true" >
       <STYLE fontSize="15" />
      </DATA_LABELS>
    </COLUMN>
   </PLOT_OPTIONS>

  <SERIES_LIST>
    <SERIES name="ST" alias="AffVersions,ST" color="#5D81E3" type="column"/>
    <SERIES name="AT" alias="AffVersions,AT" color="#D64949" type="column"/>
    <SERIES name="Production" alias="AffVersions,Production" color="#47B562"
type="column"/>
    <SERIES name="PT" alias="AffVersions,PT" color="#P64869" type="column"/>
  </SERIES_LIST>
</HIGHCHART>
```

Tento kód se týká grafu z obrázku 5.3. Major bugs reported in release. Celý graf je obalen párovým tagem <HIGHCHART>, což přirozeně označuje, že se jedná o prvek typu Java Highchart. V tagu <TITLE> je udán hlavní nadpis našeho grafu. Tag <LEGEND> slouží k nastavení parametrů a umístění legendy grafu.

Ve střední části kódu můžeme vidět tagy <X_AXIS> a <Y_AXIS>, které slouží k nastavení toho, jaký ukazatel má být vyobrazován na ose X a jaký na ose Y. V našem případě je na ose X ukazatel AffVersions, což jsou v naší databázi verze release. Osa Y pouze vyjadřuje hodnoty. Uvnitř párového tagu <PLOT_OPTIONS> můžeme nastavit specifická nastavení pro jednotlivé druhy vizualizace dat pomocí tagů <COLUMN>, <LINE>, <AREA> atd.

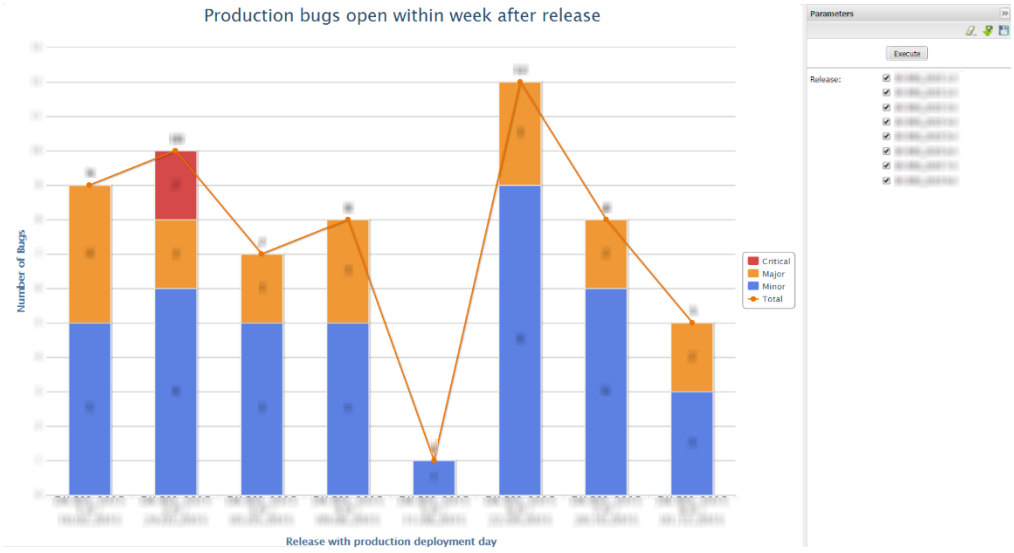
Ve spodní části kódu je párový tag <SERIES_LIST>, který obsahuje všechny metriky, jež mají být v grafu vyobrazeny. V tomto případě se jedná o počty bugů v prostředí ST, AT, Production a PT. U každé z těchto sérií je nastavena barva a typ zobrazení type="column", což indikuje, že se v případě této řady bude jednat o sloupec. Při nastavení různých hodnot type lze tímto způsobem tvořit i grafy, které kombinují odlišné typy vizualizace zároveň. Tyto grafy jsou někdy nazývány „combo charts".

Production bugs open within week after release

```
<HIGHCHART width="100%" height="100%">
  <CHART defaultSeriesType="column" zoomType="xy" />

  <TITLE text="Production bugs open within week after release">
      <STYLE fontSize="25" />
  </TITLE>
  <LEGEND layout="vertical" align="right" verticalAlign="middle" />

    <X_AXIS categories="Release" alias="Release" fontSize="30" >
        <LABELS>
           <STYLE fontSize="16" />
        </LABELS>
     <TITLE text="Release with production deployment day">
           <STYLE fontSize="15" />
     </TITLE>
    </X_AXIS>

    <Y_AXIS>
       <TITLE text="Number of Bugs">
       <STYLE fontSize="15" />
     </TITLE>
    </Y_AXIS>

    <PLOT_OPTIONS>

    <COLUMN stacking="normal" animation="true" enableMouseTracking="true"
shadow="true" showInLegend="true" visible="true">
      <DATA_LABELS  align="right" enabled="true" overflow="none"
crop="false" x="25">
```

```
        <STYLE align="left" fontSize="15" color="#000000" />
        </DATA_LABELS>
    </COLUMN>

    <LINE animation="true" enableMouseTracking="true" shadow="true"
showInLegend="true" visible="true">
        <DATA_LABELS enabled="true" >
        <STYLE fontSize="15" fontWeight="bold" color="#000000" />
        </DATA_LABELS>
    </LINE>

    </PLOT_OPTIONS>

  <SERIES_LIST>
    <SERIES name="Critical" alias="Release,Critical" color="#D64949"
type="column"/>
    <SERIES name="Major" alias="Release,Major" color="#F09835"
type="column"/>
    <SERIES name="Minor" alias="Release,Minor" color="#5D81E3"
type="column"/>
    <SERIES name="Total" alias="Release,Total" color="#F07300"
type="line"/>
  </SERIES_LIST>

</HIGHCHART>
```

V této XML šabloně můžeme vidět typický případ combo chart, jež se týkal bugů, které vznikly v sedmi pracovních dnech bezprostředně po datu posledního release. Tag <SERIES_LIST> obsahuje jak metriky s typem sloupec, tak metriku s typem linie. Typ sloupců je ještě nastaven na skládaný pomocí parametru stacking=“normal“. V tagu <CHART> je nastaven parametr zoomType=“xy“, který umožní přibližování a oddalování jednotlivých částí grafu dle potřeby. Tato vlastnost je výborná pokud chceme vyobrazit jen některá období bez zasahování do filtrace dat. Jak bude tento kombinovaný graf vypadat, můžeme vidět v následujícím obrázku 5.4., jehož plnou velikost můžeme vidět v příloze č. 4.



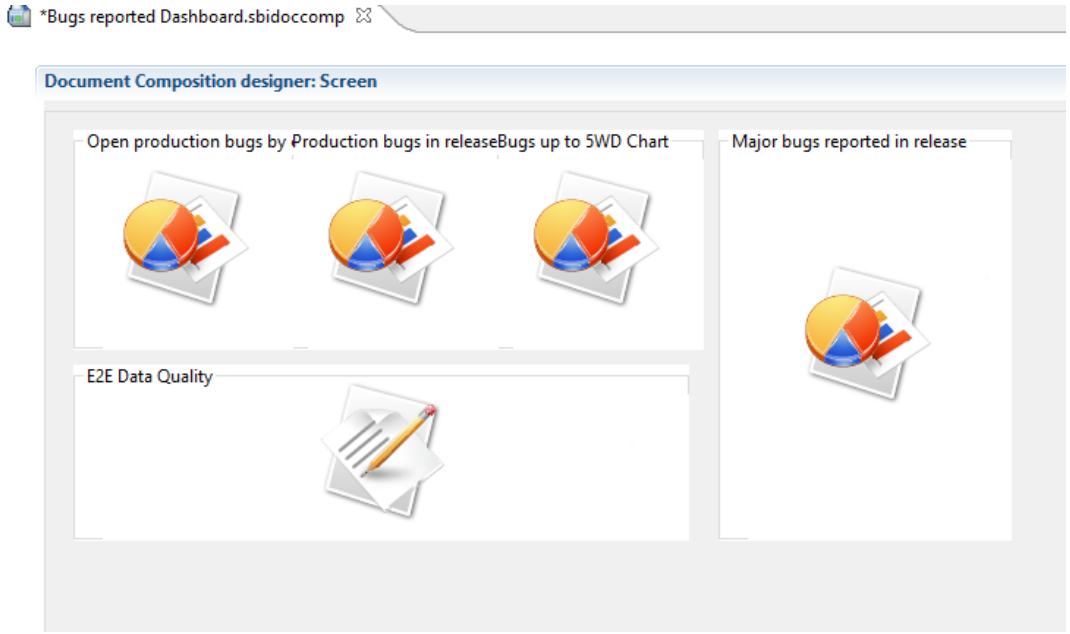
Obrázek 5.4. Production bugs open within week after release, zdroj: vlastní zpracování

Na pravé straně grafu můžeme vidět filtraci releasů řešenou pomocí checkboxů. Nicméně jak už bylo řečeno výše, samotnému filtrování se budeme věnovat až v další části. Podobně můžeme tvořit i plošné grafy nebo liniové grafy, které můžete nalézt v příloze č. 5.

5.3.4. Tvorba dashboardů

Dashboardy se ve SpagoBI nazývají Composite documents (složené dokumenty). Jejich tvorba je velmi jednoduchá a intuitivní. Nejdříve je nutno vytvořit jednotlivé dílčí reporty, grafy, tabulky nebo KPI ukazatele, které má složený dokument obsahovat. Tvoří se stejným způsobem jako grafy popsané v kapitole 5.3.3., tj. pomocí XML šablony, jež je odeslána na server a přiřadí se jí zdrojový dataset.

Poté pro tyto jednotlivé části vytvoříme objekt typu Composite document, který bude všechny prvky obalovat. Composite document je opět XML šablona, která ale nemá svůj speciální dataset, nýbrž slouží jen ke kombinování samostatných prvků.



Obrázek 5.5. Bugs reported Dashboard - návrh, zdroj: vlastní zpracování

V rámci Composite document si můžeme vytvořit libovolný počet kontejnerů pro jednotlivé prvky. Každý kontejner lze také na ploše složeného dokumentu libovolně umisťovat a měnit jeho velikost dle potřeby pomocí funkce resize. Pak už jen stačí přetáhnout do příslušných kontejnerů prvky, které mají obsahovat, pomocí jednoduchého drag and drop systému. Tato funkcionalita je velmi výhodná pro koncové uživatele. Podobný dashboard týkající se User stories můžeme vidět v příloze č. 5.

Ve finální fázi pak opět následuje deploy dashboardu na server s tím, že musíme předem mít na serveru i jednotlivé dílčí prvky, které dashboard obsahuje.

5.3.5. Filtrace

Po nahrání reportů na server už je z funkčního hlediska nutné už jen nastavení filtrace pomocí parametrů.

Vstupní parametr datasetu

Jak jsme zmínili už v kapitole selekce dat, filtrace je přímo závislá na parametrizaci selectů, které vybírají z databáze zdrojová data pro jednotlivé datasety. Poté co parametrizujeme v klauzuli where samotný select, je třeba přidat vstupní parametr v nastavení datasetu. Vstupní parametr může mít několik datových typů, ale v rámci našeho projektu se jednalo především o parametry typu string.

Tvorba seznamu hodnot

Po nastavení vstupních parametrů u datasetu jsme pracovali na vytvoření filtrace samotné. Prvním krokem je vytvoření seznamu hodnot neboli list of values (LOV). Jedná se o hodnoty, které nám samotná filtrace bude nabízet k výběru. Zde máme dvě možnosti. Hodnoty mohou být buď ručně zadány, nebo získávány pomocí selectu ze zdrojové databáze. Jelikož naše hodnoty mají být získávány dynamicky, bylo využito možnosti selektování ze zdrojové databáze. Tímto způsobem lze vytvořit i restrikce možností pro filtrování (například vybrat hodnoty jen do určitého roku nebo od určité velikosti atd.). Po vytvoření nesmíme svůj LOV zapomenout uložit.

Tvorba analytického ovladače

Analytický ovladač (analytical driver) slouží k samotnému vytvoření filtru. V možnostech analytického ovladače si můžeme vybrat, zda se má jednat o checkbox, roletku, nebo vyskakovací okno. Dále pak volíme LOV, ze kterého mají být získány hodnoty, jež se ve filtrovacím prvku vyobrazují.

Aplikace filtru

Filtr aplikujeme tak, že v možnostech nastavení reportu na serveru přidáme potřebný analytický ovladač a zadáme přesný název parametu z datasetu. Můžeme také nastavit, zda má parametr nabývat jen jedné hodnoty, nebo smí být vícehodnotový. Nakonec už stačí jen uložit nastavení reportu a při příštím spuštění reportu se nám v pravé části obrazovky objeví příslušný

analytický ovladač pro filtraci. Pokud je dataset správně parametrizován, bude report po stisku tlačítka execute reagovat na změny v analytickém ovladači tím, že vykreslí filtrovaná data.

5.3.6. Uživatelská práva

Po vytvoření všech filtrací je práce na samotném vývoji reportů dokonána. Poslední, co bylo třeba učinit, je vytvoření uživatelských účtů a nastavení příslušných práv. SpagoBI na toto myslí a už v základu nabízí několik typů uživatelských účtů Administrator, Developer, Business user a Showcase user.

Pro potřeby přístupu ze strany zákazníka nebylo třeba tvořit jiné typy uživatelských účtů. Vystačili jsme si plně s účty Administrator, Developer a Business user. Nicméně kdyby na našem serveru existovalo více různých reportingových oblastí, není problém vytvořit nespočet typů uživatelských účtů tak, aby každý uživatel měl přístup jen k určité oblasti reportingu nebo jen k určitým dashboardům.

6. Závěr

Cílem práce bylo zkvalitnění reportingu pro jednoho ze zákazníků firmy Tieto, pomocí automatizovaného dynamického reportingu. Byl kladen důraz na co nejnižší náklady celého řešení, proto bylo řešení realizováno pomocí Open Source BI nástrojů. Byly také zadány požadavky na online přístupnost k reportingovému serveru, automatizaci celého reportingu a na vzhled a formát realizovaných reportů a dashboardů, korespondující s dosavadním vzhledem zákazníkova reportingu. Všechna citlivá zákaznická data byla v práci znehodnocena a znečitelněna, jelikož firemní politika zákazníka jejich zveřejnění nedovoluje.

Metodická část byla věnována teoretickým východiskům business intelligence, stavbě BI systému a využití BI v rozhodovacím procesu. Pozornost byla věnována také nástrojům pro získávání a zvyšování kvality dat, datovým uložištím a datovým pumpám ETL a ELT. Poslední podkapitola se zabývala reportingem a vizualizací dat, v souladu s cílem práce.

V rámci praktické části byly nejprve formulovány podmínky zadání projektu a informace o zadavatelské firmě Tieto. Poté byla provedena analýza současného stavu zákazníkova řešení. Následovalo zhodnocení možností několika OS BI nástrojů dostupných na trhu a výběr tří potenciálních kandidátů pro splnění požadovaných funkcionalit. Po vyhodnocení dotazníku vyplněného třiceti BI specialisty z Tieto, byl vybrán finální nástroj SpagoBI. V konečné fázi

byl navržen datový model, sestrojeno ETL řešení pomocí Talend Open Studio a proveden návrh a realizace uceleného dynamického reportingového řešení pomocí SpagoBI.

Výsledné řešení odpovídalo parametrům a požadavkům, které byly zadány a diplomová práce byla tudíž splněna v celém svém rozsahu. BI řešení reportingu je plně funkční a splňuje všechny zadané funkcionality, ba i více. Samotný reportingový server je pro uživatele přístupný a je již zákazníkem plně využíván.

Během výzkumu bylo zjištěno, že na trhu existuje nemalé množství OS BI nástrojů, které jsou v praxi reálně využitelné. Pracnost OS řešení je sice vysoká, podpora menší a některé kroky a postupy jsou o poznání složitější než v placených nástrojích, ovšem co se výsledku týče, OS řešení je schopno placeným nástrojům mnohdy konkurovat. Pokud je tedy zákazník ochoten tolerovat pracnost vzhledem k nízkým nákladům, dá se pro firmu vytvořit efektivní BI řešení s minimálními náklady na software. Obzvláště pokud se jedná o menší nebo střední firmu jsou funkcionality OS nástrojů více než dostačující.

Práce se zabývala řešením reportingu a dynamických vizualizací především z interních firemních dat, avšak díky poměrně rozsáhlým funkcionalitám OS nástrojů pro správu big data by bylo možné prakticky využít OS BI nástrojů i pro integraci a zpracování big data. Funkcionality z této oblasti jsou v poslední době neustále více a více žádané a nástroje pro jejich efektivní řešení často nebývají nejlevnější. Proto by se využití OS nástrojů pro zpracování big data mohlo jevit jako velmi zajímavé řešení. Nicméně nejprve by bylo třeba provést patřičný výzkum a podrobné srovnání dostupných možností v oblasti zpracování big data pomocí OS nástrojů.

Seznam použité literatury

Tištěné knihy

BIERE, Mike. *The New Era of Enterprise Business Intelligence: Using Analytics to Achieve a Global Competitive Advantage*. Boston: Pearson Education, 2011. ISBN 978-0137075423.

INMON, William H. *Building the Data Warehouse.* 4th ed. Indianapolis: Wiley, 2005. ISBN 978-0-7645-9944-6.

KIMBALL, Ralph a Margy ROSS. *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling.* 3rd ed. New York: Wiley Publishing, 2013. ISBN 11-185-3080-2

LACKO, Luboslav. *Databáze: datové sklady, analýza OLAP a dolování dat.* Brno: Computer Press, 2003. ISBN 80-7226-969-0.

LOSHIN, David. *Business intelligence: the savvy manager's guide, getting onboard with emerging IT.* Boston: Morgan Kaufmann Publishers, 2012. ISBN 15-586-0916-4

VERCELLIS, Carlo. *Business Intelligence – Data Mining and Optimization for Decision Making*. Indianapolis: Wiley Publishing, 2009. ISBN 978-0470511398.

WITHEE, Ken. *Microsoft Business Intelligence For Dummies*. Hoboken: Wiley Publishing, Inc., 2010. ISBN 04-705-2693-9

Internetové zdroje

AANDERUD, Tricia. *BI Dashboard: Tips for Learning to Use the Tool.* [online]. 2012, [cit. 2016-02-03]. Dostupné z: http://bi-notes.com/2012/09/bi-dashboard-tips-for-learning-to-use-the-tool/

Ariacom. *6 major reasons to use SEAL REPORT.* [online]. 2016, [cit. 2016-02-09]. Dostupné z: http://www.sealreport.org/default.cshtml

DIKUBE. *Let Knowledge derived from your Data drive your Organization and we at DiKube can supply the road map.* [online]. 2011, [cit. 2016-02-02]. Dostupné z: http://www.dikube.com/service_im_consolidation.php?action=tab_di

ECKERSON, Wayne a Mark HAMMOND. *Visual Reporting and Analysis: Seeing Is Knowing.* [online]. 2011, [cit. 2016-02-03]. Dostupné z: https://tdwi.org/research/2011/01/bpr-q1-visual-reporting-and-analysis

Jedox AG. *Business-Driven Intelligence: Easy. Fast. Flexible.* [online]. 2016, [cit. 2016-02-09]. Dostupné z: http://www.jedox.com/en/bi-company

LAUMANS, Joel. *An Introduction to Visualizing Data.* [online]. 2009, [cit. 2016-02-03]. Dostupné z: http://www.babaksohrabi.com/Files/TextNews/AnIntroductionToVisualizingData.pdf

Pentaho Corporation. *ENTERPRISE-CLASS AND OPEN SOURCE HERITAGE.* [online]. 2016, [cit. 2016-02-09]. Dostupné z: http://www.pentaho.com/about

SCHILLER, Martin. *Co se skrývá pod zkratkou ETL?*. [online]. 2003, [cit. 2016-02-02]. Dostupné z: http://www.systemonline.cz/clanky/co-se-skryva-pod-zkratkou-etl.htm

SMITH, Jeffrey. *Data Warehouse Architecture.* [online]. 2013, [cit. 2016-02-02]. Dostupné z: http://data-warehouses.net/architecture/

SPEARE, Geoff. *ETL vs. ELT – What's the Big Difference?.* [online]. 2015, [cit. 2016-02-02]. Dostupné z: https://www.ironsidegroup.com/2015/03/01/etl-vs-elt-whats-the-big-difference/

SpagoBI Labs. *100% Open Source Business Intelligence and Big Data analytics.* [online]. 2016, [cit. 2016-02-11]. Dostupné z: http://www.spagobi.org/

Talend. *Talend Open Studio.* [online]. 2016, [cit. 2016-02-18]. Dostupné z: https://www.talend.com/products/talend-open-studio

The Eclipse Foundation. *What is BIRT?.* [online]. 2014, [cit. 2016-02-04]. Dostupné z: http://www.eclipse.org/birt/about/

Tibco Software, Inc. *JasperReports Library.* [online]. 2016, [cit. 2016-02-09]. Dostupné z: http://community.jaspersoft.com/project/jasperreports-library

VAVRUŠKA, Jindřich. *ETL a kvalita dat.* [online]. 2003, [cit. 2016-02-02]. Dostupné z: http://www.systemonline.cz/clanky/etl-a-kvalita-dat.htm

Seznam zkratek

BI – Business Intelligence

CEO – Chief Executive Officer

CSS – Cascading Style Sheets

DM – Data Mart

DSA – Data Staging Area

DW – Data Warehouse

DWA – Data Warehouse Appliance

EE – Enterprise Edition

ELT – Extraction, loading, transformation

ETL – Extraction, transformation, loading

GYI – Graph Your Inbox

HTML – Hypertext Markup Language

IT – Informační technologie

JDK – Java Development Kit

JDBC – Java Database Connectivity

JSON – JavaScript Object Notation

KPI – Key Performance Indicator

MOLAP – Multidimensional Online Analytical Processing

MS – Microsoft

ODS – Operational Data Storage

OLAP – Online Analytical Processing

PDF – Portable Document Format

ROLAP – Relational Online Analytical Processing

SP – Scatter Plot

SSIS – SQL Server Integration Services

SW – Software

TOS – Talend Open Studio

XML – Extensible Markup Language

Prohlašuji, že

- jsem byl seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. – autorský zákon, zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo;

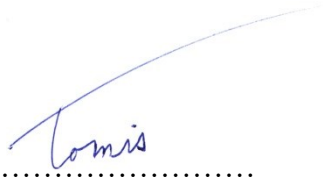
- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen VŠB-TUO) má právo nevýdělečně, ke své vnitřní potřebě, diplomovou práci užít (§ 35 odst. 3);

- souhlasím s tím, že diplomová práce bude v elektronické podobě archivována v Ústřední knihovně VŠB-TUO a jeden výtisk bude uložen u vedoucího diplomové práce. Souhlasím s tím, že bibliografické údaje o diplomové práci budou zveřejněny v informačním systému VŠB-TUO;

- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu § 12 odst. 4 autorského zákona;

- bylo sjednáno, že užít své dílo, diplomovou práci, nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).

V Ostravě dne 22. dubna 2016



.....

Tomáš Tomis

Adresa trvalého pobytu studenta:

Na Sedlácích 1011

739 34 Šenov

Seznam příloh

Přílohy